PROYECTO FIN DE CARRERA

TEMA: Transmisión de video en Internet

TÍTULO: Research and Production of a Multi-view Streaming Page.

AUTOR: Teresa Aragón Tribaldos y Verónica Cabezas Urgilés
TUTOR: Jürgen Lohr
DEPARTAMENTO: Fachbereich VI: Informatik und Medien
CENTRO DE LECTURA: Beuth Hochschule für Technik Berlin

Fecha de Lectura: 20. 9. 2012

Calificación: 1,3

RESUMEN DEL PROYECTO:

Los medios de comunicación electrónicos y mecánicos como las películas, la televisión, la fotografía o el offset son sólo ejemplos de la rapidez y la importancia del desarrollo tecnológico al que ha llegado la sociedad. Sin embargo, las tecnologías punteras y el continuo desarrollo han proporcionado posibilidades cada vez más destacadas para tener servicios reformadores. Hoy en día el multi-video ha sido desarrollado con diferentes herramientas y aplicaciones, teniendo como principal objetivo ser más avanzadas y proporcionar nuevas opciones accesibles para todos los usuarios, en general en términos de gestión y accesibilidad (sólo se necesita conexión a internet). El propósito de todas las tecnologías es generar una innovación con el fin de adquirir más usuarios y adquirir popularidad en el mercado. Debido a esto, es importante realizar una implementación en este caso. Bajo esos términos destacando el alcance que tiene el multi-video y la importancia de ser parte de la globalización actualmente conlleva a que surja una aplicación que de la posibilidad de seleccionar el lenguaje bajo un mismo escenario. Por último es importante destacar que gracias al continuo desarrollo de tecnología del multi-video, un nuevo y ampliado mercado intercultural será alcanzado haciendo de esto un crecimiento contiguo con el desarrollo global en la actualidad.

BEUTH HOCHSCHULE FÜR TECHNIK BERLIN

University of Applied Sciences

Final Bachelor Thesis

Research and Production of a Multi-view Streaming Page

Verónica Cabezas Urgilés

Teresa Aragón Tribaldos

Sound and Image

Year 2011-2012

Escuela Universitaria de Ingeniería

Técnica de Telecomunicación

Universidad Politécnica de Madrid

Proyecto Fin de Carrera (Plan 2000)

Research and Production of a Multi-view Streaming Page

Verónica Cabezas Urgilés

Teresa Aragón Tribaldos

Sound and Image (Sonido e Imagen)

Year 2011-2012

Abstract

The electronic and mechanical media such as film, television, photography, offset, are just examples of how fast and important the technological development had become in society. Nevertheless the outcoming technologies and the continuous development had provided newer and better possibilities every time for having advanced services. Nowadays Multi-view video has been developed with different tools and applications, having as main goal to be more innovative and bring within technical offerings in a friendly for all users in general, in terms of managing and accessibility (just internet connection is needed). The intention of all technologies is to generate an innovation in order to gain more users and start being popular, therefore is important to realize an implementation in this case. In such terms realizing about the outreach that Multi View Video, an importance to become more global in this days, an application that supports this aim such as the possibility of language selection within the use of a same scenario has been realized. Finally is important to point out that thanks to the Multi View Video's continuous progress in technology a more intercultural market will be reachable, making of it a shared society growth on the world's global development.

Content

1 Chapter 1

Introduction and objectives

1.1 Introduction

With the emergence of Internet, different and new ways of communication have been explored to reach the maximum audience possible. Video and audio are one of the main communication sources available on the Internet. First, they were downloaded, which usually requires high bandwidth. Nowadays, thanks to streaming, it is possible to visualize videos and audio in real time without downloading the content, using less bandwidth than older technologies.

Several technology trends have combined to make video transport over IP networks useful for a wide variety of applications today, such as learning applications.

Learning systems are in constant evolution since its inception, growing exponentially in their facilities and possibilities of use. With the advent of electronic design and even more with the massive use of Internet, a whole new way of learning has been developed in the form of online education. Nowadays, it is desirable that any learning system possesses the property of ubiquity, which means that learning can be done from anywhere, anytime and from any device.

1.2 Objectives

Based on these concepts, the objectives of this thesis are the study of the video element linked to Internet, the new developments of the Multi-view video, especially linked to educational resources.

First, a Single-view scenario will be developed in which a single video will be supported, explaining the different phases until the final product. Second, the Single-view scenario will be developed into a Multi-view scenario, a website that will handle multi-streaming videos (5) on which the viewers can interact. The five videos will be shown on a sidebar where the viewer can choose the video he wants to see on a bigger screen, while having absolute control over the player controls.

Thinking of possible applications of this Multi-video scenario, such as a learning tool, another tool with language selecting option (German, English and Spanish) will be developed for this thesis.

The following chapters are developed concepts of video on the Internet, from the used technologies, including the transmission protocols to the future implementation of the video on the Internet with HTML5. Particularly emphasized is the streaming process, because its distribution and consumption have grown exponentially in recent years. In the last chapters, an economic analysis will be shown, in order to show out the profitability of this product development in the market, which studies the profitability, marketing, sale and future lines of video streaming focused on virtual learning.

2 Chapter 2

The Video on the Internet

Internet is changing: Static content is giving way to streaming video, text is being replaced by music and the spoken word, and interactive audio and video is becoming commonplace. These changes require new applications, and they pose new and unique challenges for application designers.

Many Internet users consider the Internet as a technology of relationship. In its current position as mass media, the video plays one of the most important roles. All video editors have their own objectives, including business communication, education and entertainment. Below are some examples.

2.1 Companies and organizations

Companies of all sizes use videoconferencing. It is a way of communication between professionals all around the world that allows business to happen without displacement. Another example is Media Vest, the advertising giant that recently pulled millions of dollars from its TV advertising budget to invest in online video advertising.

Also digital newspapers and political parties use video on the Internet to transmit information and share ideas. In fact, many politicians have their own blog or YouTube videos.

2.2 Education

The e-learning platforms are using multimedia technology as teaching material. Also universities use videos as a way for broadcasting lectures and events.

2.3 Entertainment

If somebody misses an episode of “Breaking Bad” or the latest U2 concert, it is possible to enjoy a video on the own PC via YouTube for example.

YouTube is the website that has developed the corner on the market in streaming video. This site has grown exponentially in the past few years, globally distributing millions of memes and videos posted by Internet users.

3 Chapter 3
State of Art

Before the streaming technology appeared, Multimedia content playback over the Internet necessarily implies having to completely unload the wrapper file to the local hard drive. As audio and video files tend to be huge, the download and access as complete package becomes a very slow operation.

However, with the streaming technology, a file can be downloaded and reproduced at the same time, making the waiting time minimal.

Streaming media usage has grown exponentially over the past few years, both for entertainment purposes and as a vehicle for organizations to market, sell, and support their products and services, as well as for internal communications and training. For many such organizations, streaming video has transitioned from a curiosity to a mission critical technology.

At the same time, streaming media technologies have the challenge of reaching the full range of target viewers. Thus, there are emerging new alternatives for distributing streaming media.

Streaming technologies like Adobe Flash, Apple QuickTime, Microsoft Windows Media and Silverlight, all include certain common components in their solutions. These include a player to play the media on the viewer's computer or mobile device, a defined file format or formats that the player will play, and often a server component that offers features like digital rights management and live streaming.

All streaming technologies use compression to shrink the size of the audio and video files so they can be retrieved and played by viewers in real time. Common video compression technologies, also called codecs, include H.264, MPEG-4, VP6 and VP8, Windows Media Video (WMV), and MPEG-1/2, while common audio codecs include AAC (Advanced Audio Coding), Vorbis, Windows Media Audio (WMA), and MP3.

Today, a media stream can be streamed either live or on demand. Live streams, or also called true streaming, send the information directly to the computer or device without saving the file to a hard disk. On Demand streaming is called progressive streaming or progressive download, which means that the file is saved on a hard disk and then played from that location.

Due to significant growth in streaming media, organizations that distribute streaming media must continually monitor and evaluate market trends to ensure that they choose the technologies that deliver the highest quality streams possible for their target viewers.

3.1 Development of Streaming Media

The evolution of streaming can be viewed in three separate time lines, one relating to streaming technologies, the next to connection speed and the target device, and the final relating to the growth of streaming service providers.

3.1.1 Streaming Technology

On June 24, 1993, „Severe Tire Damage” was the first band to perform live on the Internet. The band was playing a gig at Xerox PARC, while scientists were talking about new technologies for broadcasting media on the internet using multicasting. As proof that their new technology worked out, the band was broadcasted live over the Internet. In that way, people could see the concert live over the Internet in Australia and elsewhere.

Real Networks was another pioneer in the streaming media market and broadcasted one of the earlier audio events live over the Internet. This was a baseball game between the Yankees and the Seattle Mariners in 1995. They launched the first video streaming technology in 1997 with RealPlayer. According to some sources, by the year 2000, more than 85% of streaming content on the Internet was in Real format.

The problem was that Real depended on the sale of servers, while Microsoft and Apple were at the top of the market. In addition, consumers started to avoid the free Real Player, which to that time was the default player for all the multimedia content, because it continuously bothered users with upgrades. The once so essential player was now being uninstalled and also not installed on newer computers.

3.1.1.1 *The Microsoft Era*

In the beginning of 1996, Microsoft developed a media player known as ActiveMovie, which allowed to stream media and also contained a proprietary streaming format. This was the predecessor to the streaming feature that was available in Windows Media Player 6.4 in the year 1999.

From the early 2000 until around 2007, Window Media dominated the computer panorama, giving Windows Media Player a dominant share of available desktops and notebook. This was the most widely used format on the Internet and in most company intranets. However, Windows Media Player had a non-customizable interface and depended upon a third party plug-in to play on the Mac, which was just starting to begin its revival.

At the same time, web design was transitioning from HTML to Flash, which offered much greater interactivity and design flexibility. Though Flash had a video component, the initial codecs offered poor video quality and incomplete audio/video synchronization. That changed when in 2005 Macromedia licensed the On2 VP6 codec.

3.1.1.2 *The Flash Era*

Via VP6, Macromedia (and then Adobe, who acquired Macromedia in 2005), could match Microsoft's video quality in a player that could be integrated with the rest of a Flash-based site, which was truly cross platform and near ubiquitous. This proved irresistible to most broadcast and entertainment sites as for example YouTube. The Microsoft's share in the markets dropped to single digits by 2010. Thus, to stop the boom, Microsoft released in 2007 a Flash competitor called Silverlight, but the player never achieved the necessary success to convince Flash users to adopt it.

Then, in April 2010, Apple shipped the iPad, which didn't support Adobe Flash. The technology that Apple uses for video playback on its iOS devices is HTML5. Rather than using a plug-in based player like Flash, Windows Media or Silverlight, HTML5 uses a player native to the browser to playback the video file.

3.1.1.3 *The Rise of HTML5 Video*

HTML5 is the latest specification of HTML (hypertext markup language), the language used to create websites. The specification is currently under review by the World Wide Web Consortium (W3C) and Web Hypertext Application Technology Working Group (WHATWG), and is set to be finalized in 2014. By the start of 2011, the W3C and WHATWG announced that the specification would be henceforth known as HTML; however in this thesis, it will be referred as HTML5.

For HTML5 videos to work, the user must have an HTML5-compatible browser, and that browser must support the codec used to compress the file. In early 2011, only about half of the available browsers were HTML5 compatible. That changed with the release of Internet Explorer 9, even though in a nearly future, HTML5 will not achieve the 96% penetration enjoyed by Flash.

Furthermore, codec support within those browsers was different. Mozilla Firefox, Google Chrome, and the Opera browser support Google's WebM, but not H.264. Apple Safari and all iOS devices are compatible only with H.264. Internet Explorer 9 comes with an H.264 player, but not with a WebM player. This difference in codec support means that organizations seeking to be completely HTML5 compatible must support at least two codecs. The HTML5 technology does not offer yet features like adaptive streaming, digital rights management, live broadcasting, and many others that are proven components of Flash, Silverlight, Windows Media, and QuickTime. Advanced features like peer-to-peer delivery and multicasting, launched by Adobe with Flash Media Server 4, are not yet among their possibilities.

In short term, HTML5 is most appropriately viewed as a solution for iOS and perhaps for some mobile devices, but not as a replacement for Flash or Silverlight in general-purpose computers. However today, web sites can certainly implement HTML5 using Flash as a fallback, if the viewer connects with a non-HTML5-compatible browser. And any site that extensively uses Flash or Silverlight should at least be thinking about how to integrate HTML5 support into their future plans.

3.1.2 Connection Speed and Target Device

When in 1997 streaming video was launched, the target market included desktop and notebook users. Today, the range of targets is much more diverse, including low-power mobile devices, set-top boxes, and powerful computers connecting via DSL or cable, many of which are connected to HDTV monitors. To reach these devices, streaming producers must customize streams for delivery via 3G (or slower) networks as well as 50Mbps broadband.

The disparity in playback power and connection speed has spawned technologies like adaptive streaming, where a single video is encoded into multiple data rates and sizes, spanning the target viewer community. A stream, appropriate for playback devices and connection speed is transmitted and adaptively adjusted to changing connection conditions and available CPU.

Often, these target devices support different streaming technologies, as with iOS devices with HTML5 and computers with Flash. To support the expanding range of viewers, many web sites support multiple streaming technologies, with many producing adaptive Flash streams for computer playback and H.264 streams for iOS devices.

With the mobile devices that are dependent on battery life, such as tablets and smart phones, the development of digital media streaming is now focused on formats that do not depend on Adobe Flash, known for its relatively high computer resource usage and thus compromising a mobile device's battery life.

3.1.2.1 *From DIY to OVP*

At the beginnings of the streaming media market, the thought for virtually all companies who distributed streaming media was "do it yourself" (DIY). That is, companies encoded their videos, created their players, hosted their streams, and maintained the streaming servers that distributed the streams.

However, the streaming media market has become much more complicated, with competing technologies like HTML5 and Flash, and a much more diverse universe of connection speeds and playback platforms. This complexity has increased both the CAPEX and personnel cost of effectively delivering streaming video.

As a result, many organizations are turning to online video platform (OVP) providers like Brightcove, Sorenson, and Ooyala. Operationally, the videos are uploaded to the OVP, which encodes them into the necessary formats to serve to the target viewers. The OVP provides customizable players for a range of target platforms and embed codes for integrating the player into the web site. It also hosts the necessary servers for delivering the streaming media to the consumer and provides media management and analytics packages.

Moreover, some companies use User-generated content (UGC), sites like YouTube and Vimeo for a subset of these services. For example, many of the videos presented on IBM's website are hosted by YouTube and the results are amazing. As of early March 2011, on the IBM Smarter Planet YouTube channel, IBM had accumulated more than 3,757 subscribers and more than 2 million video views. The most popular video, "IBM and the Jeopardy Challenge," had been viewed 613,849 times. And the cost of the YouTube channel is a tiny fraction of television advertising. This approach has multiple profits, including decreased CAPEX and personnel costs associated with hosting the video. In addition, since YouTube is developing their player for millions of streaming producers, they can quickly incorporate new technologies.

3.1.3 Streaming Media Market Landscape

The streaming media market is influenced by a series of companies in diverse sections, each with their own power to adopt or reject different technologies. Certainly, some companies participate in multiple sections and exert influence on them.

3.1.3.1 *Streaming Media Technology Providers*

The streaming technology providers are the most influential group, which chooses the technologies and services to integrate into their platforms. These include Apple, which provides QuickTime as well as the HTML5-based technology to reach iOS devices, Adobe with Flash, and Microsoft with Windows Media and Silverlight. Although W3C is not a company itself, it helps to set the standards for HTML5, including the supported audio and video codecs.

3.1.3.2 *Streaming Media Playback Platforms*

The owners of the various playback platforms decide, which of the commercially available technologies to use. This became obvious with Apple's decision, not to support Flash on the iOS platform or WebM in its Safari browser.

In the early days of streaming, the most relevant playback platforms were Windows and Macintosh computers. While Apple and Microsoft still hold great influence, the mobile devices comprise the fastest growing sector of streaming media viewers. These mobile devices tend to be split between hardware vendors like LG, Samsung, Motorola, Nokia, HTC and mobile operating system providers like Google (Android) and Microsoft (Windows Phone 7). In particular, Google and Microsoft have to create a mobile platform, deemed as marketable by the hardware vendors that select it. This leads to interesting juxtapositions, like Google promoting WebM and HTML5 in its Chrome browser and on YouTube, while integrating Flash into its Android operating system.

Browser vendors like Microsoft (Internet Explorer), Mozilla (Firefox), Google (Chrome), Apple (Safari), and Opera (the Opera Browser) also influence which technologies will get adapted. During the early days of the HTML5 debate, Mozilla refused to integrate an H.264 player into its browser. That resulted in the need of a high-quality alternative that is satisfied by WebM. Now, Apple's intransigence is preventing WebM from becoming the unified HTML5 standard.

Also in terms of volume and influence, set-top box providers like Roku, Boxee, Apple and Syabas will increasingly affect technology decision-making, as their use expands.

3.1.3.3 *Streaming Media Delivery Providers*

Delivery providers like OVPs and UGC sites also influence streaming technology adoption. For example, though Microsoft introduced Silverlight in 2007, it wasn't supported by any OVP until 2010, stunting its adoption. In contrast, OVPs like Brightcove and Kaltura and UGC sites like YouTube and Vimeo were among the first to support the iPad and HTML5, accelerating their adoption. The key OVPs include Brightcove, Kaltura, Ooyala, Sorenson Media, Powerstream, and ClickStreamTV, while the most notable UGC providers are YouTube, Vimeo, Viddler, DailyMotion and Metacafe.

It takes wide bandwidth to deliver HD-sized streaming video, which is why most high-volume streaming publishers and OVPs use content delivery networks (CDNs) like Akamai, Limelight, Global Crossing or Level 3 to distribute their videos. Though they are in second plane, there are multiple CDNs of all sizes, and have a significant impact on technology adaption. The high-volume Flash streaming wouldn't be possible without CDNs installed in Flash Media servers. Many CDNs quickly adapted the technology that enables adaptive streaming to Apple iOS devices, helping to make that channel viable for streaming producers. Without that support, the appearance of iOS devices as a potential market would have been completely different.

4 Chapter 4
Video Transmission over Internet

Today, transporting video signals is a world business. Whether for entertainment, education, or personal communication, we now live in a world where we are exposed to video content in many forms. The scale of the technologies, systems and protocols that are used to gather and deliver all of this content is amazing and continues to evolve.

The media files are streamed over the general telecommunications network. Most streaming files are delivered over a data network. It may be the local network or, for an enterprise with widely dispersed sites, a wide-area network. The Internet has become ubiquitous for data communications, from simple e-mail to complex electronic commerce applications.

Internet video is used to supply video content to viewers by way of the public network. In a typical Internet video installation, service providers set up a website portal that can be reached by anyone with a standard browser. At this site, a list or index of the various pieces of content will be available. Once the user has selected content, it is delivered from servers to the viewer’s PC, where media viewer software can be used or where it can be downloaded to another device.

For correct communication to occur between users and servers (client-server architecture), the data transmission has to fit a set of rules or protocols that will be presented in the following chapter.

4.1 The OSI reference model

The concept of interconnected networking, or the Internet, has its origins in the quest by the U.S. military to connect research institutions over a packets switched network. In the 1970s the U.S. Department of Defense developed the DARPA project, a multilayer model of network protocols that evolved into today's Internet. The International Standards Organization later augmented the communication protocols, which evolved into the Open Systems Interface model (OSI). The OSI reference model provides a useful basis for discussion and comparison of layered systems.

The Internet does not wholly adhere to the open systems interface. The Figure 1 shows the relationship, but note that the principles are similar. Later protocols do adhere more closely to the OSI seven-layer model.

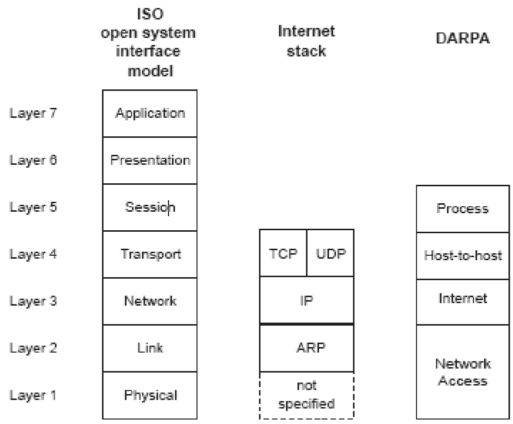


Figure 1 Multilayer network model

The model comprises a set of seven layers. At each layer in the model, there is a logical communication between that layer on one host and the equivalent layer on another. When an application on one system wants to talk to an application on another system, the communication proceeds down through the layers at the source, passes over the physical connection, and then go up the protocol stack to the destination.

For example, a Web browser application renders an HTML presentation, which is delivered using an HTTP session, over a TCP transport connection, via an IP network, over an Ethernet data link, using twisted-pair physical cable. The result is the transfer of a Web page from application (Web server) to application (Web browser).

In the following paragraphs, the most important protocols for video transmission, in which the network and transmission layers take place, will be described.

4.2 Network layer

The network layer (layer 3) is a level which connects links, unifying them into a single network. It provides addressing and routing of messages. It may also provide control of congestion in the switches, prioritization of certain messages, billing, and so on.

4.2.1 IP

Internet Protocol, or IP, is the main network communication protocol. The other protocols at layer 3 are used for control of the network routers to set up the connections.

Internet Protocol is a protocol not oriented to connection, used by both, the source and the destination, for data communication over a packet-switched network.

Internet information is transmitted in small chunks called "packets" or “datagrams”, which are sets of data sent as independent messages. Each package can be transmitted on a different way, depending on the congestion of the routes. The important thing is to rebuild the message emitted at the destination, not the path taken by packets that compose it.

The idea of using packet networks to transport voice and video is not new, it goes back to the early 1970s. The first RFC on this subject, the Network Voice Protocol (NVP), dates from 1977. Video came later, but still there is over ten years of experience with audio/video conferencing and streaming on the Internet.

IP has an unreliable datagram service. It has drawbacks like the variable network latency, the probable arrival of packets in a different order from transmission and the possible packet loss. IP doesn´t provide a mechanism for determining if a packet reaches its destination or not. It only provides security of the headers but not of the data transmitted. If reliability is needed, it is provided by the protocols of the transport layer, such as TCP.

IP is the common element in today's Internet. Perhaps the most complex aspects of IP are addressing and routing, therefore IPv6 is currently used. The Internet Protocol version 6 (IPv6) is a version of Internet Protocol (IP), designed to replace Internet Protocol version 4 (IPv4). The main benefits of this version are the higher number of available addresses.

“The Internet Protocol provides the abstraction of a single network, but this does not change the underlying nature of the system. Even though it appears to be a single network, in reality the Internet consists of many separate networks, connected by gateways, now more commonly called routers, and unified by the single service and address space of IP.”

The following figure shows a very simplified way of a possible Internet video network.

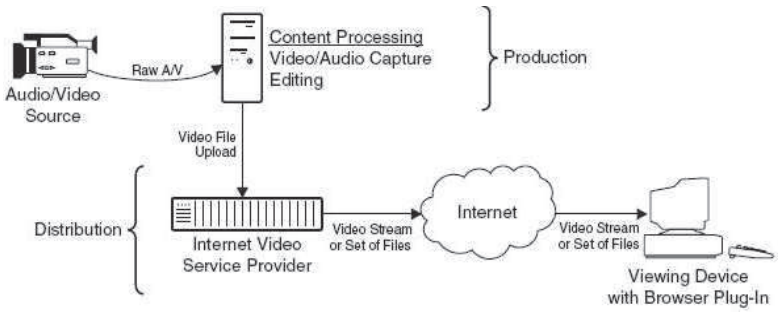


Figure 2. Typical Internet Network

4.3 Transport layer

The transport layer (layer 4) is the first end-to-end layer. It takes responsibility for the delivery of messages from one system to another, using the services provided by the network layer. This responsibility includes providing reliability and flow control if they are needed by the session layer but not provided by the network layer.

4.3.1 TCP/IP

Transmission Control Protocol, or TCP, is one of the fundamental Internet protocols.TCP and IP are used together, forming TCP/IP.

It is a connection-oriented protocol and as stated above, one of the great strengths of TCP is its reliability. The built-in error protection of TCP makes it an excellent protocol for the delivery of general purpose data, but the way this is implemented proves to be a disadvantage for streaming applications. TCP sequences the data bytes with a forwarding acknowledgement number that indicates the next byte the source expects to receive, to the destination. If bytes are not acknowledged within a specified time period, they are retransmitted. This feature of TCP allows devices to detect lost packets and request a retransmission. The repeated transmission will add to the communication latency, but that is not normally an issue with data exchange. TCP also provides flow control of the data.

With audio and video, the viewer requires a continuous stream to view the source in real-time. Retransmission of data is going to add delays; retransmission also uses up bandwidth in the data channel.

TCP supports many of the popular Internet applications (browsers, file sharing, FTP clients, ...) and application protocols HTTP, SMTP, SSH and FTP.

4.3.2 UDP/IP

User Datagram Protocol, or UDP, is another fundamental Internet protocol. Here as well, UDP is used together with IP, forming UDP/IP.

It is a connectionless transport mechanism that can support high-speed information flows, such as digital video and many other types of data transport. It is frequently used when the overhead of setting up a connection (as is done by TCP) is not needed. For example, UDP is often used for broadcasting messages from one device to all the other devices on a network segment; for example, if a printer is out of paper, the print server will alert all the users who use this printer that it is out of paper.UDP is a connectionless protocol, which means there is no mechanism to set up a connection between a sending host and a receiving host. The UDP sender simply formats datagrams with the correct destination socket (IP address and port number) and passes them to IP for transport. Of course, this also means that there is no coordination between a UDP data transmitter and a UDP data receiver to ensure that the data is transferred properly.

With this lack of coordination, it could seem that UDP is unsuitable for video data transfer, since missing video data can interfere with the receiver's ability to display a correct sequence of complete images. But there are many video stream formats, which also include a mechanism for detecting and correcting errors. For example, MPEG transport streams can include bytes for Reed-Solomon forward error correction. When these bytes are included, the MPEG decoder can correct bit errors and can sometimes re-create lost packets, making it unnecessary to retransmit them. Also the media players can often mask video data errors. When these capabilities are available, UDP is a logical choice for the transport protocol, since it does not add unneeded overhead to streams that already have built-in error-correction functions.

4.3.3 UDP vs. TCP

If a stream gets interrupted, it will result in the interruption of the video playback. This may cause loss or distortion of one or several video frames, but that is a transient event, which will be ignored by the viewer.

For real-time applications, delivery on time is more important than error-free transmission, which means, for our purposes that we give up the reliability in exchange for timeliness. Therefore, the most appropriate protocol for streaming video transmission is UDP.

5 Chapter 5

Streaming Video

5.1 Definition

Streaming video is a process that sends the content in a compressed form over the Internet and is displayed by the viewer in real time. Using streaming media enables the viewer to watch the content as it arrives, the data is sent in a continuous stream of multiplexed packets, instead of waiting until the file is completely downloaded. The user needs a player, which is a special program that uncompress and sends video data to the display and audio data to the speakers. A player can be either an integral part of a browser as a plug-in or a program on the viewer´s computer.

Streaming video is usually sent from prerecorded video files, but can be distributed as part of a live broadcast "feed". In a live broadcast, the video signal is converted into a compressed digital signal and transmitted from a special Web server that is able to do multicast (sending the same file to multiple users at the same time).

In the following chapter we will explain how a streaming process works, its architecture and its main applications.

5.2 Types of Streaming

A streaming process can be divided into two types, according on how the media was obtained. These two types are Video on Demand and Live Streaming.

5.2.1 Video on Demand (VoD)

Video-on-Demand (VoD) is an interactive multimedia service which enables users to start watching the video of their choice at the time they want.

VoD allows users to control operations such as play, pause, fast forward and rewind, while downloading the video in parallel.

Nowadays, the amount of titles available on the web is comparable with the amount of titles available in average video stores. The storage of the media files can take the form of a centralized server, equipped to send media simultaneously to hundreds of viewers or it can take the form of more distributed storage throughout the network.

Various forms of VOD have been tried over the years; the next table lists the most popular types of VOD services

Table 1. Types of VoD Services

Type	***Description***
True Video on Demand	This is the purest form of VOD, where each viewer receives an individual video stream over which he or she has complete control. Viewers are allowed to start, stop, pause, rewind and fast-forward the content. Typically a fee has to be payed e for each title viewed.
Near Video on Demand	The difference to true VOD, is the absence of the individual video stream control capabilities. An example of NVOD is called stagger casting, in which multiple copies of a program are played, starting at five-minute intervals.
Subscription Video on Demand	Is the same technology and viewer control as VOD. The difference is the payment system. In SVOD, subscribers pay monthly a fixed fee for unlimited access to a library.
Free video on Demand	A variation on VOD where payment is eliminated. In most systems, this content is restricted to long-form advertisements, how-to guides and other low-cost content.
Personal video recorders	These devices take incoming video programming, compress it and record it to a hard disk that is typically located either in an STB or a stand-alone device. Viewers then control the PVR to play back content. Viewers normally program the PVRs to record specific programs at specific times.
Network personal video recorders	Offers functionality similar to PVRs, but recording is performed inside the service provider's network rather than in the viewer's location. Some content owners contend that this technology is so similar in capability to true VOD that it needs to be licensed as such.
Pay per view	This precursor technology to VOD is used primarily to deliver live paid programming, such as concerts and sports events.

5.2.2 Live Streaming or IPTV

Live streaming transmits media events in the same moment in which they take place. For example, the transmissions of a concert, a match or e- learning classes usually are transmitted at the same moment of the recording. The transmission of radio and Internet television also has these characteristics, although occasionally some of the information is not part of a live event, for example, a program that has previously been recorded, but is transmitted at a determined time.

Unlike in VoD streaming, all the users watching the same channel at the same time would have their viewing times synchronized. The only operation that the users should be able to perform with Internet television is to switch channels.

For this type of transmission the term broadcast is used, because the same information is transmitted "live" to all customers. Thus, regardless of when a client connects to a server, everybody watches exactly the same point of the stream in a determined instant (except the variations in network delays that make some customers to receive the data before others).

This type of streaming requires, besides the conventional elements for streaming transmission, an element that carries out the process of capturing and compressing the media in real time (sometimes known as broadcaster). This equipment can be installed on the same machine as the streaming server, if the number of potential customers is not big. But for professional results in an environment with many clients, it is suitable to separate the two programs on two different machines. In addition, to achieve an efficient service in this type of transmission, it is desirable that the streaming process is carried out with multicast techniques.

5.3 Architecture of a streaming network

A general streaming architecture can be defined by different stages. Each stage involves different devices and applications, the most important ones are the streaming server, whose main responsibility is to deliver streams to each user's device, the media player software that generates the images on the user's display and the content and the transportation network between the server and the viewing device.

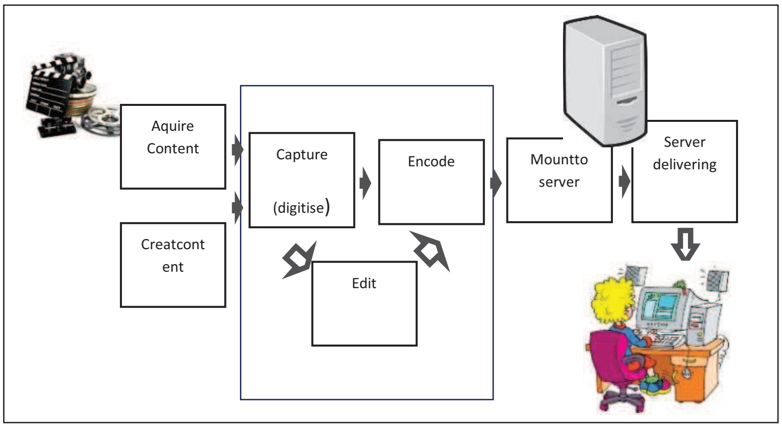


Figure 3. A typical streaming architecture

5.3.1 Content Preparation

The content to be transmitted, such as a live image generated by a camera or a video previously recorded, always needs to be processed. The main processes, including capturing the image by the camera are:

- ***Capture***: Recording the video content.
- **Editing**: The process of editing a movie is also seen as the second directing, because through the post production it is possible to change the intention of the movie. Editing turns individual scenes, called raw footage, into a finished motion picture. Editors splice all of the usable footage together into a coherent storyline according to the storyboard.

  The process is basically always the same; first the images are loaded onto a computer using a program. With this program, it is easy to try different ways of arranging the shots, mainly in the order specified by the shooting script.
- ***Preprocessing***: Adjust the video for the compression process. That includes color correction, noise removal, image resizing, and others.

5.3.2 Encoding the video

Encoding the video is the stage where the video signal is converted into a streaming media file.

The encoding process takes place in several stages. The first one is to convert the video signal into a file that computers can process. The second step is to reduce the data rate, by scaling and compressing the bit rate. The third and last step is to wrap the compressed video in a packetized format that can be streamed over IP network.

The usual way to encode is first to convert the video signal into an intermediate format that can be handled as a computer file, usually the format chosen is AVI. Then the AVI video is converted into a streaming format by the encoder, which uses a software codec.

5.3.2.1 *Codec*

Codec means Compression/decompression. Codecs are programs that use mathematical algorithms to convert the video files into compact streaming media formats.

Decompression stage takes place in the end user's computer. It can be a compatible decoder or, most times, is part of the media player itself.

Streaming codecs are usually asymmetrical, which means that the encoding process requires much more processing than the decoder.

The selected codec depends on which ones are playable on the different browsers and its version. There are no standard codecs. In the following table are formats listed, which are supported by the various browsers. In order to reach the biggest possible audience, usually various streaming files with different codecs are generated.

Table 2. Compatibility between codecs and browsers

Feature	Chrome	Firefox	Explorer	Opera	Safari
Basic Support	3.0	3.5	9.0	10.5	3.1
<audio> WAVE, PCM	?	3.5	--	--	3.1
<audio>WEbM, Vorbis	Yes	4.0	--	10.6	3.1 (must be installed separately)
<audio>Ogg, Vorbis	Yes	3.5	--	10.5	3.1 (must be installed separately)
<audio> MPEG4, MP3	?	--	9.0	--	3.1
<audio> MPEG4, AAC	?	--	9.0	--	3.1
<audio>Ogg, Opus	?	15.0	?	?	?
<video>WebM, VP,Vorbis	?		9-0	10.6	3.1 (must be installed separately)
<video>Ogg, Theora, Vorbis	Yes	--	-	10.5	3.1 (must be installed separately)
<video>MPEG4, H.264, MP3	?	--	9.0	---	3.1
<video> MPEG4, H.264, AAC	?	--	9.0	---	3.1
Any other format	--	--	--	---	3.1 Plays all formats via Quicktime

5.3.2.2 *Lowering the bitrate*

The reduction of the bit rate has to be achieved in a way that it still produces a recognizable picture. The bit rate usually is reduced by a big factor such as 4,000. There are two methods to achieve this reduction, first by scaling the video data, which can achieve a reduction of over 130:1; second by using a compression algorithm, to remove redundant data.

5.3.2.3 *Scaling*

There are three techniques to scale down the original video size.

- ***Spatial scaling:*** This technique reduces the video frame size. Typical resolutions are 320x240 or 176x144 pixels.
- ***Temporal scaling:*** Reduces the number of frames per second. In some cases the frame rate can be reduced to the half or more, but when there are scenes with fast movements like a sportscast can be seriously affected by this reduction. Therefore some encoders use a variable frame rate; in that case a maximum frame rate must be determined.
- ***Color resolution:*** The last method reduces the bit depth of each sample. In RGB, each color is represented by 8 bits, including the padded for processing, which builds a word of 32 bits. It can be reduced to 16 or 8 bits; depending on the subject of the video; continuous tone objects will show a posterization effect when a lower color resolution will be used.

  By using the YUV signal, the color information can be sampled with a lower rate, because the human visual system primarily uses the luminance information. The color sampling rate can be sized down by the half (4:2:2) or by one quarter (4:2:1), depending on the performance required.

5.3.3 Compression

Compression removes the redundant perceptual information. There are two methods of video compression, special (intraframe) and temporal (interframe) compression.

- ***Intraframe Compression:*** This method is used to compress single frames. Simple algorithms such as run-length encoding or a lossy system (original data cannot be completely recovered) can be used.
- ***Interframe Compression*:** With this method the information is discarded when it does not change from one frame to the next one. The transmitted information refers to the areas where the picture has changed. This compression is used by the MPEG-1, MPEG-2 and MPEG-4 standards.

5.3.4 Encapsulation

In this process, the compressed video file has to be formatted into IP packets, the headers and other data required to comply with a specific protocol have to be added, then the packets can be transmitted over the Internet. During the next step, control data gets wrapped around the audio and video data, so the streaming server can control real-time streaming.

This process is usually done in real time, just before the packets are sent over the network, because information on the headers usually changes for each user (IP address destination).

The software used to encapsulate the streaming files is included in different devices or programs such as the encoders and file servers.

5.3.4.1 *IP Packets*

IP works as a delivery service for IP packets. A packet is a single message unit that can be sent over IP, with very specific format and content rules.

An IP packet is formed by a header, which contains special information such as destination, version, etc., and by a piece of data content, which can have different lengths between a maximum and minimum. The format of the header and data is shown in the Figure Nº 4.

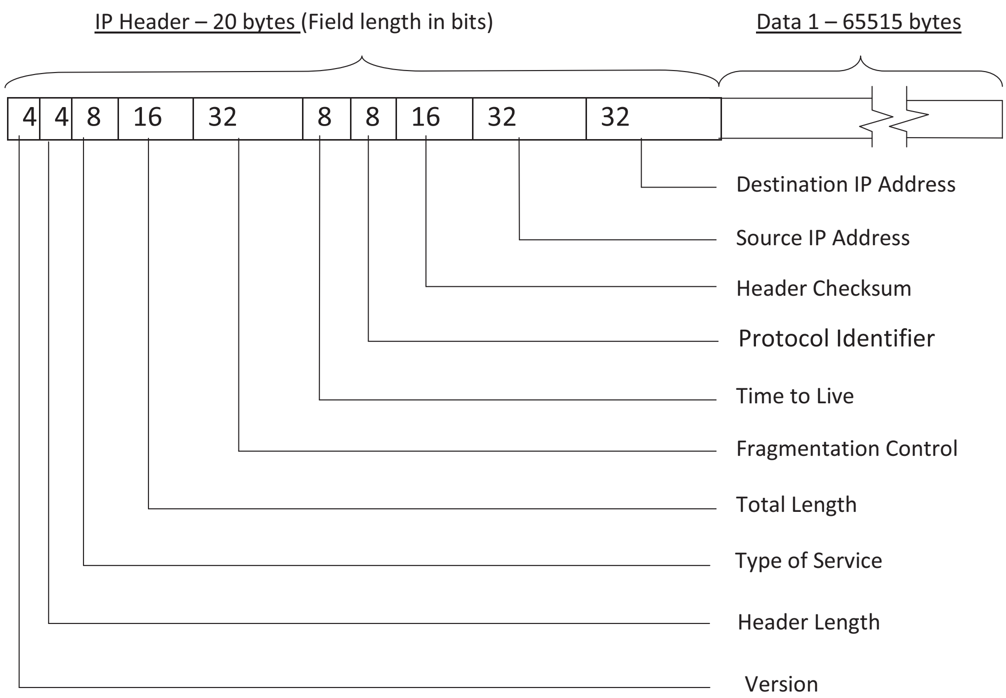


Figure 4.Format of an IP Packet

5.3.4.1.1 Packet Sizes

The length of the packets is specified by a minimum and a maximum on the IP specifications. This specification does not include the header, because all packets have the same header bytes. Choosing the length of a packet has no recipe, but it has to be considered that the size of the packets affects the performance of the video signals.

Here are some advantages of long and short packets

5.3.4.1.1.1 LONG PACKETS

- Less overhead. With long packets, it will be less header information as a percentage of the over-all packet size; this means that less bandwidth is required by long packets.
- Reduced packet processing load. Each time a packet arrives, some process, such as examining the header of the packet has to occur. With long packets, the amount of work is reduced.
- Greater network loading. Gaps have to be sent between packets, with longer packets, fewer gaps are required; therefore, more bandwidth can be used for the data.

5.3.4.1.1.2 SHORT PACKETS

- Lost packets are less harmful. When a packet suffers a bit error in its header, it is discarded, so is its video data. Since the packets sent are short, less information is lost, and this can be covered by the device, so the viewer does not notice it.
- Reduced latency. The packets are sent only when they are full, short packets are filled faster than longer packets
- Less need for fragmentation. When a packet is too long, it has to be fragmented. This means that it has to be broken down in small packets, which can produce overhead for the router and other devices. The best action is to choose a size smaller than the maximum established.

To minimize the percentage of overhead, the longest possible packet sizes that won't result in fragmentation on the network are used for video applications.

5.3.4.2 ***MPEG STREAM TYPES***

MPEG stream types are not the only ones used nowadays; each browser has its specific video format. MPEG stream types are defined according to international standards. Also, some of the other compression systems use the same standards. There are three different stream types.

5.3.4.2.1 Elementary Stream

Elementary Streams are the raw outputs from MPEG video and audio encoders. An Elementary stream contains only one type of content, so if there is audio and video, there are two elementary streams, one for video and one for audio.

Elementary streams are also formed by special data that is used by the decoder to determine how the data should be decoded, such as video frame type, position of each data block on the video screen are indicated, etc.

5.3.4.2.2 Packetized Elementary Streams

A packetized elementary stream (PES) is simply an elementary stream that has been broken up into easy-to-handle "packets."

Note that PES packets are of variable lengths and can be a hundred kilobytes. Each packet of a PES will have a header, which includes a code number that indicates the source of the elementary stream. When audio and video from different elementary streams are combined, we can figure out which video and audio signals belong together.

Some of the PES headers include time stamps. For video signals, two different kinds of time stamps can be used; presentation times stamps (PTSs) and decode time stamps (DTSs). The PTS of a frame indicates the time at which it should be presented or displayed to the viewer as a video signal. The DTS of a frame indicates the time at which it should be decoded by the MPEG decoder.

5.3.4.2.3 Program Streams

Program Streams are formed by several types of packetized elementary streams to support production and recording tasks. A common clock is also added for the synchronization.

This means that a Program stream contains "a program" which is a combination of video, different audio streams, and related data such as alternate language sound tracks and commentaries. Data can include captioning information, information about the program stream (title, length, owner, content descriptions, etc.), or control information, needed by recording or playback devices.

A single program is made up of one or more packetized elementary streams and can have added data and control information. Program streams are commonly used in DVD.

Program streams can be of variable bit rate and typically use large packets.

5.3.4.2.4 Transport Streams

Transport streams (TSs) are another way of combining several packetized streams into a single entity, like program streams. But the difference is that TS can be transported across the network, the different packetized streams can be from different clock sources. Transport stream packets have a defined length.

Transport streams use length fixed packets of 188 bytes. Each packet contains data from only a single elementary stream video, audio, data, or control information, but not a mixture. Forward error correction (FEC) codes can be added to transport stream packets through the use of standardized codes such as Reed-Solomon (RS).

It is important to note that IP packets will normally contain multiple transport stream packets, usually seven, because seven is the highest number of transport stream packets that can fit into a 1500-byte IP packet, which can be carried across an Ethernet network without fragmentation.

5.3.4.3 *Program Clock References*

A program clock reference (PCR) is a 33-bit number that must be inserted periodically into every transport stream to allow an MPEG decoder to synchronize with the 27-MHz clock, used by the encoder when the MPEG compression was done. Synchronization must be maintained to make sure that the decoder doesn't run out of data to decode (buffer underflow) and that it doesn't receive more data than it can handle (buffer overflow).

Periodically, the encoder copies the value of this counter and places it into a transport stream packet. At the decoder, these counter values are extracted and used to create a synchronized copy of the encoder's clock. The accuracy of the decoder's clock depends on a smooth, steady flow of data from the encoder to the decoder. Any changes in the amount of time it takes for the packets to travel through the network (known as delay variation or packet jitter) need to be minimized.

5.3.5 Streaming Server

There are mainly two ways in which streaming can be delivered; first, as pull process and second, as a push process. The former one is described by the television model. That means that it is used to stream live or prerecorded content as a webcast. This is why push streaming can be used for web channels or live events. The user can alternatively pull prerecorded content on-demand and this is kind of similar to using a CD-ROM or a web browser. In the case of interactive content, the client or player requests files from a server while with simulated-live webcasts, the server runs a playlist, which is able to stream files at a scheduled time to the player.

Table 3.Web Server versus Streaming Server

	Web Server	Streaming Server
Advantages	• Part of an existing infrastructure. No additional training or expertise for IT staff necessary.	• Optimized media delivery. • Dynamic stream control. • Interactive media control. • Multicast support. • Improved Server Hardware Utilization. • Supports Live Web casting.
Disadvantages	• None of the streaming server advantages. • Supports only progressive download.	• Additional equipment is required.

In the following, we are going to describe what a streaming server is. Nowadays, the web server, which is typified by Apache, is the most-used server for delivering multimedia content. To deliver HTML pages and their associated image files, these web servers use preferably HTTP over TCP/IP and the latter is used as the transport layer over the internet. The files are downloaded as fast as the system allows to the web browser cache.

5.3.5.1 *Adapting to network congestion*

For maintaining the correct delivery rate over UDP/IP (or TCP/IP if bandwidth permits), the Real-Time Protocol is used. The client interactions with the stream, the VCR-like controls (Play, Pause and so on) are supported by the RTSP framework. To measure network congestion and to switch between stream rates for multiple bit rate media files, RTCP reports from the player are usually used by the streaming server application. Because of this, the player can report lost and delayed packets, and also the reception of out-of-sequence packets.

5.3.5.2 *Loading Content*

The first step you should consider when you want to use a managed service or want to do your own service is to deliver your content to the streaming servers. The encoding most likely takes place near a video editing facility or, in the case of a live webcast, at the venue. If you are only streaming over a local area, no Internet backbone is needed, while in other cases the servers have to be located close to an Internet backbone. That means that in a normal case, the encoder and server are separated geographically. File transfer (e.g. FTP) is the easiest way to deliver the content. Some encoding systems are able to transfer files automatically, directly after the encoding has finished.

Live Streaming:

The file can be sent on a CD-ROM, but neither of the mentioned methods is suitable, if the content is a live broadcast. Then it has to be streamed. There are two ways in which a media encoder connects to the server. By using TCP for bidirectional control links and UDP for unidirectional media streams. The circuit used for this connection, needs a more than sufficient bandwidth and a high QoS; that means low packet loss and timing jitter. All the receivers of the webcast are able to see any data loss or corruption. That means to preferably use an uncontended circuit like T-1/E-1, rather than a domestic circuit like ADSL or a cable modem.

5.3.5.3 ***Announcing the content***

Embedded in a web page is a hyperlink, which contains the URL for the content and the instruction to start the media player. It is this hyperlink through which the player locates the streaming media.

5.3.5.4 ***Playlists***

Streaming regular webcasts means to play out clips to a prepared schedule. Clips can be put in playlists and streamed at the right moment. Distance learning and corporate presentations can be scheduled at fixed times. The viewer can watch the desired stream at the appropriate time. SMIL can be used to program the playlist. Apple QuickTime Streaming Server has a user interface for creating and editing playlists, which can be used for MPEG-4 or audio-only MP3 playlists.

5.3.5.5 ***Server deployment***

5.3.5.5.1 Video server hardware

To give real-time playback, the streaming server has to read the media files from disk, packetize them and deliver them at the correct rate. These tasks have to be performed for multiple concurrent streams. Asynchronous operation in a regular office server application contrasts with real-time requirement. At times, where the demand of resources is the highest, file delivery is delayed in the former case. In the case of a video stream, this would result in stalling playback. But with time it is becoming easier and easier to meet the demands of streaming, because the performance of server components is improving.

The areas to focus include:

- Disk drive performance
- Network card performance: It can be advantageous to use multiple cards for streaming and control
- The system and I/O bus bandwidths
- Symmetrical multiprocessor CPUs

- The stripping out of unnecessary software services, leaving the resources for uninterrupted stream delivers.
- System memory large enough to manage multiple high-speed streaming buffers

It is necessary to determine the true capacity of a server configuration, which requires certain tests and measurements. The basic configuration can be calculated from the bandwidths. Rather than using one big multiprocessor server, it is advantageous to use many small servers and to scale wide. This makes the system more tolerant to faults. A single server can get lost from the cluster, without losing the whole site. After determining the available bandwidth and how many users will connect to the system, you can decide on a server design. It could be a different directory or you can use a separate staging server to test content before it is placed on the public server.

5.3.5.5.2 Hosting

The easiest way to realize high-performance streaming to large numbers of public clients is to use a hosting facility or a content delivery network, because hosting providers usually have very wide band connectivity to the backbone. Service providers are usually located in secure buildings, with standby power facilities, halon fire extinguisher systems and multiple paths to the internet backbone. It is these factors which make outsourcing superior, especially from the point of view of physical security.

If files contain confidential information, it might be better to have the servers on own premises managed by local staff. If you decide to use DRM with pre-encryption, this may not be much of an issue. Turnkey hosting, shared hosting and co-located hosting are the different services offered by the companies. The simplest way to implement is turnkey hosting. Media files are uploaded, using FTP and the rest is done for you. Co-location gives you more control. You can install your own server plant in secure cages that you rent. The advantages are that you can monitor your servers from your own location by telnet and you get reliable power and connectivity. Before taking a decision, think precisely about the services you really need and study the service level agreement (SLA) carefully.

5.3.5.5.3 High availability

The set-up of a high availability system requires the system to be secure, reliable and easy to maintain. A fault-tolerant design is required to achieve a higher level of system availability. This might come in two different forms. First in the form of redundant hardware (power supplies, fans, NICs) and second in the form of redundant multiple server architectures. Hot-pluggable components like disk drives, power supplies and fans are also desirable to minimize downtime.

5.3.5.5.4 Security

Security is a very important issue, especially if you want to stream around the clock and it becomes a core business service. Security is determined by many factors. On the one side are physical threats like fire, theft and power outages. The best ways to deal with the former two are redundant mirrored server sites, which are able to deal with those issues. A reputable hosting service will cover most of these issues too, but check the service level agreement. On the other hand, there are the hackers, which, if they can gain access to your system, can wreak havoc or use denial of service attacks. The best way to deal with these threats is to call in a network security consultant to advice and audit your systems.

5.3.5.5.4.1 Authentication and authorization:

Authentication is used to verify the identity of a client while authorization allows authenticated clients to access confidential information. There are three types of users (system administrators, content creators uploading content, viewers of the content), which gain access to the system by a password. Some form of digital rights management provides better protection against unauthorized access to your content, this is especially important, if your files are confidential or of commercial value.

5.3.5.6 ***Reflecting Server***

A reflecting server takes one copy of a video stream and makes multiple copies for delivery to multiple users in real time. In this manner, it can add streams to a live event when demand peaks above the capacity of the original streaming server.

Users will have their connections redirected during session setup so that they will receive their stream from the nearest reflecting server that has available capacity.

5.3.6 IP Streaming Network

5.3.6.1 ***Internet Transport***

There is no such thing as a perfect network, with no delays or a huge bandwidth, etc. But in the following table we will list the desirable values for the properties a network should have for good streaming experiences.

Table 4. Desirables values for a good network

Feature	
High bandwidth	The public Internet certainly has a large amount of bandwidth; however, there are also plenty of users. No single user can count on having a large amount of bandwidth available at any given time for a video application.
Low jitter	Jitter is difficult to control in private networks and almost impossible to control in the public Internet. Because there is no universal mechanism to ensure that packets from a video stream all follow the same route, some jitter and packet reordering are inevitable.
Low delay	Network delay depends on two things: the time it takes a signal to physically travel over a link and the time it takes to process packets at each step of their journey. In the public Internet, users do not have control over how their packets are routed. This means that the delay can be long or short, depending on the route used and the congestion level of the devices along the route.
Priority control	The public Internet provides essentially no priority control, because the Internet connects all users equally. There is no reliable mechanism to reserve bandwidth for a specific user or to make specific packets for one user to take priority over other traffic on the public Internet.
Lossless Transmission	Overall, the public Internet is extremely robust; communications can take place even if major portions of the network are not operational. However, packet loss is a fact of life on the Internet: Some carriers promise only 99 or99.5% packet delivery, meaning that losses of 0.5 or 1% of a user's packets are not unusual.

5.3.6.2 *Sending a Packet*

The process of sending a packet from one computer to another may involve transit through many different IP network links. At each step along the way, the IP packets must be processed. This processing involves the following procedures:

The header of the IP packet must first have its header checksum verified. If the checksum is incorrect (due to a bit error on the data link, for example), then the packet is destroyed.

The header is then examined to determine where the packet is going, by looking at the destination IP address. Based on this IP address, the network equipment (typically an IP router) determines what to do with the packet, such as send it out on another telecom link or transfer the packet to a local area network connected to the equipment.

The time-to-live counter is decremented, a new checksum is calculated, and both are inserted into the packet header in place of their former values. If the time-to-live counter reaches zero, then the packet is destroyed. (This prevents packets from endlessly circulating around a network.)

5.3.6.3 ***Transport considerations***

There are some factors that have to be considered, when video is transported over an IP network. These factors have to be considered because they can affect the user's experience.

5.3.6.3.1 Multiplexing

Sometimes, the video distribution delivers video streams combined from different sources, if this process is used, a multiplexer is needed to process the video signals.

Multiplexing is useful because:

- Larger streams are easier to transport and administrate than several smaller streams.
- The combinations of variable-rate streams, especially when the peaks in one stream correspond to the valleys of the other, allow using the bandwidth more efficiently.

On the other hand, multiplexing can add a small amount of delay to the video signals.

Two types of multiplexing are commonly used today, Time division multiplexing and statistical multiplexing.

5.3.6.3.1.1 Time division multiplexing

Incoming packets are placed in the necessary timeslots in the combined stream. This method provides a fixed amount of bandwidth to each incoming (tributary) stream.

As a main advantage of using this method is that it is simple and generates low overhead. Therefore it is efficient for fixed rate tributary streams. Another benefit is that it provides constant end-to-end delay through a network.

5.3.6.3.1.2 Statistical multiplexing

The bandwidth is provided to incoming channels in response to their changing rates; higher-speed tributaries are dynamically given a larger amount of the overall network capacity.

This multiplexing method is suitable for tributaries with data rate peaks at different times, which happens in compressed video. That way the bandwidth is used more efficiently.

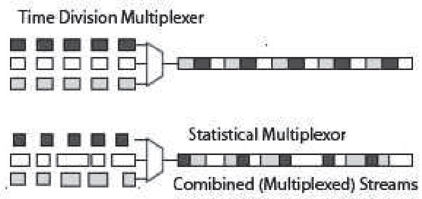


Figure 5. Types of multiplexing

Multiplexing should be used when multiple signals need to be delivered to the same set of destinations and also when the network costs are high.

5.3.6.3.2 Buffering

Buffering is the data preload into a reserved area of memory, in streaming; it refers to downloading a certain amount of data before starting to play.

In video, especially MPEG, buffers are required, which are used on the encoding and decoding phases to support the motion estimation, this is the comparison between frames to see what moved from one to the other, or to help the frame reordering.

Buffering also supports error correction because packets need to be processed to add forward error correction (FEC) coding.

On the receiver's side, the buffer is used to smooth out the variations in data arrival times, caused by statistical multiplexing and switching.

Large buffers on the receiver's side support the needed error correction or they de-interleave the packets. This means that, if a packet gets lost and won't get recovered, the buffer can smooth it. In case the packet arrives, the buffering gives time to reinsert the lost packet in the proper position.

Large receiver buffers are almost universally used in IP streaming media applications. This type of buffers can hold up to 10 or 20 seconds.

5.3.6.4 *Network Impairments*

Many things can happen between the video source and the video viewer. Some failures result from the network itself, for example cable cuts. But there are also some subtle failures, called impairments. These impairments can affect the video network, which can be the difference between success and failure.

5.3.6.4.1 Packet Loss

A packet is considered lost, when it does not arrive at its destination, or when it arrives with a big time delay that cannot be relocated.

The reasons for this problem can be a network failure, a wrong configuration, which sends the packets to other addresses or subtle errors, such as a bit error in the packet header.

Another reason why packets can get lost is due to the saturation of the network with traffic.

Lost packets would affect a small portion of the audio or the video because we do not have the information of them. One of the methods used to prevent this, is called COP3 ROW-COLUMN FEC.

5.3.6.4.1.1 COP3 ROW-COLUMN FEC

This powerful mechanism has been standardized as SMPTE 2022 COP3.

Row column FEC arranges groups of packets in rows and columns and then adds an FEC packet to each row and each column. If one packet in a row or column is lost, then its value can be calculated by performing a XOR on all the other packets in the row or column, including the FEC packet.

Bust errors can be corrected of up to 5 packets in length, and also replace complete packets.

The main consequence of adding row and column FEC is the additional bandwidth required and since more packets are sent, the system will have more delays. It is also necessary to have more buffers available, to process the higher income of packets and to check the FEC, to correct any errors in the data block.

5.3.6.4.2 Packet reordering

Sometimes, packets arrive in a different order from the one they were sent. This means, they have to be reordered and in the case of video data streams, these out-of-order packets can cause problems.

Streaming video data, particularly MPEG video packets, have a very precisely defined structure and sequence. This means, they always have to be in order. The timing of this data is also important. If for example, some data of an image is not available and it is time to display that particular image, then it is possible that some parts of this image are missing or that the image is not displayable at all.

The main reason, why packets can get out of order in a network, is network congestion. Packets might get delayed while waiting for their turn to be sent or for automatic retransmission in a wireless link.

Packets that arrive out of order can easily be put back in order through the use of a buffer. However, this can add delay to the signal. Plus, if the buffer is not large enough, a severely delayed packet will be treated as a lost packet.

5.3.6.4.3 Delay

One of the main problems on streaming video is the delay. There are two main sources of delay on IP networks, propagation delay and switching delay.

- Propagation delay is the amount of time it takes for information to travel from one location to another.
- Switching delay can appear at any point in the network, where a signal needs to be switched or routed

5.3.6.4.4 Jitter

Jitter measures the variation in the arrival time of data packets. With an ideal network, the time between packets is always the same; that means no jitter. But in reality, receivers must be produced in a way that they can tolerate jitter, and networks should be designed in a way that they won't create a lot of jitter if they are used to transport video streams.

However, for real-time data such as video, audio, and voice signals, timing is very important and excessive jitter makes receiver clock synchronization harder to achieve.

On the receiver's side, buffers can be used to reduce the jitter, by smoothing out the packet timing and by giving late packets more time to arrive.

Since always, buffering is needed and nowadays, some systems use adaptive buffering. This means, that the buffer is kept small and the delay short, if the network conditions are good. If the network conditions become worse or are poor, the size of the buffer will increase. With any video stream, clean, uncongested networks with minimal packet loss and low jitter are the best choice for transport.

5.3.7 Player Software

It is the player software, which accepts the streaming content on the user's PC. Once the content is delivered to the user's device (PC), the software player has to convert it into a displayable image.

The player can be usually used as a plug-in to a web browser but there is also the alternative of a stand-alone player. These players have been developed into media portals with navigation to popular material and often offer unique content, specific to a given architecture.

Figure 6. Player as computer software

The designers of the players have made them robust and foolproof to deploy. They also have to be easy to install or control, since it is desirable to have the largest audience as possible.

An ideal player should have the following characteristics:

- Free of charge
- Automatic installation
- Invisible software upgrades
- Compact software executable
- No configuration
- Modest use of processor resources

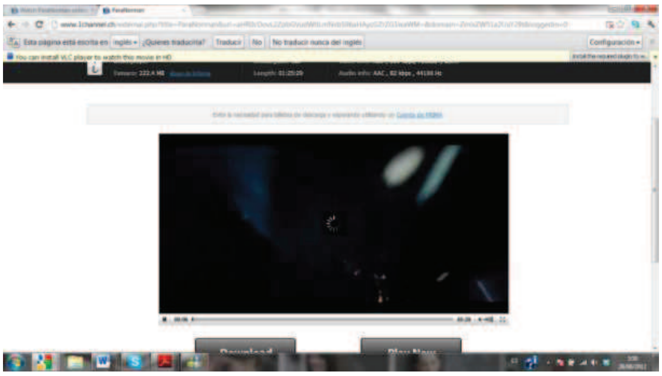


Figure 7. .Player as a plug-in to a web browser

To begin with the streaming session, the user has to choose first from a list of a web page that is hosted on a streaming server, and the user simply clicks on the appropriate hot link to begin the content playback. If the streaming server has encrypted the content, the player software needs to decrypt the incoming packets. The player software needs to know the key. It can be obtained directly from the streaming server or by connecting to a third-party authentication service.

The streaming media is sent in two sessions, one for audio and one for video as is said by the protocol RTP. Therefore, the player software is responsible for re-synchronizing the incoming streams. This is accomplished by looking at the time-stamp data contained in each stream and comparing it to the time-stamp data contained in the associated RTCP overhead packets.

After decrypting the incoming signal, the player software has to decompress it and create an image to display. The amount of processing required varies, depending on the size of the image and on the compression method used. Older compression systems (such as MPEG-1) tend to be easier to decode than newer systems (such as MPEG-4) and therefore, place fewer burdens on the decoding device. Smaller images, with fewer pixels to process are also easier to decode.

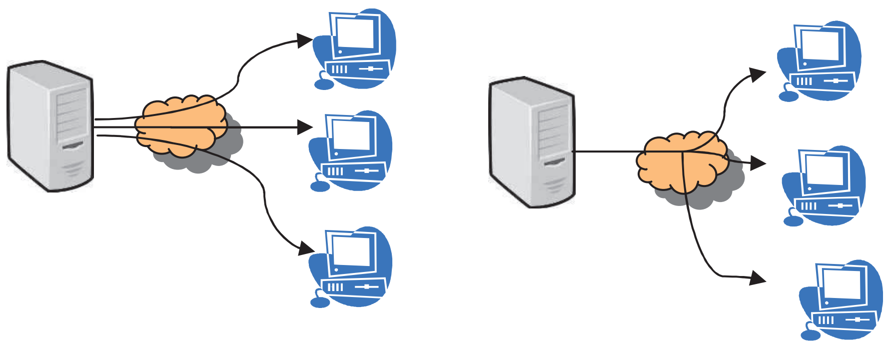
Generally, disk drive capacity is not a big factor, since the buffers required for video processing are usually held in memory rather than on disk. In some cases, for high-quality full-screen video decoding, a hardware decoder card to improve performance can also be added to a personal computer.

5.4 Multicasting

In multicasting, a single video signal will be send to multiple users at the same time. That means that like in traditional television broadcasting, all the viewers get the same signal at the same time. But for video streaming and Internet video, multicasting is more of an exception than the rule.

5.4.1 Basic concepts

In the following, we will compare unicast with multicast. Like we already said, in multicast, many users receive a single video stream at the same time. In unicast, each video stream is sent to just one recipient. If many viewers want to receive the same video, the video source has to create a separate unicast video stream for each one of those viewers. These streams are then delivered from the source to every single viewer over the IP network. In multicast, it is the network, where copies of the video stream are done for every viewer, not the source like in unicast. The process of copying occurs inside the network and there, only at the points where they are needed.



Unicast Multicast

Figure 8. Ip Unicasting vs. Multicasting

5.4.2 Benefits of multicast

- In unicast, the source has to make copies of the video signal for every single recipient. In multicast, the source sends just one video signal over the network. Inside the network, the video signal is copied for every recipient. So there is a greatly reduced burden on the source in multicast.
- The amount of data sent between locations is reduced, because just one video signal has to be sent to the network.
- In unicasting, the source has to know the address of each recipient. This is not the case in multicasting, where the source sends one video signal to the network.

5.4.3 Unicasting

Usually, in unicasting, packets are sent from the source to a single destination over an IP network. The packets are sent through the network after a request to the video source, every single packet has the destination IP address. The source has to prepare separate packets for each destination, when the data to be sent is the same for multiple destinations. Unicasting requires a significant amount of processing power and also a network connection with high bandwidth. Because the more viewers there are, the more streams of packets the source has to create. On the other hand, unicasting allows each viewer specialized features such as pause, rewind and fast-forward video. This is a big advantage which does not exist in multicasting.

Table 5. Advantages and Disadvantages of Unicasting

Advantages	Disadvantages
Unicasting works on standard IP networks, including the public internet.	The video source must have enough processing capacity and network bandwidth to create a stream for each user. When the source´s bandwidth is fully consumed, no more users can be added.
Each user has an independent videos stream. The video source can offer each user playback controls, such as pause, fast-forward and rewind. Users don’t need to wait for regularly scheduled broadcasts.	There must be enough bandwidth in every segment of the network to deliver all the streams from the source to their final destinations.
The source can determine which unicast destinations are allowed to receive the data stream and keep records of each recipient.	The video source must know the correct IP address of every active viewer´s device.

5.4.4 Multicasting

In multicasting, the opposite is the case. Here the network has the major burden of creating streams for every recipient. Inside the network, IP routers are able to recognize packets that are being multicast and to send them to multiple destinations, because of specialized protocols. Those protocols give the packets special multicast addresses. Another special protocol allows devices to join the multicast. Many routers support multicasting, but because of the burden multicasting places on processing resources, this function is not enabled in many networks. Multicasting works one-way that means that multicasts operate just in one direction and any interactivity between the endpoints and the video source must be handled by another mechanism.

Table 6. Advantages and Disadvantages of Multicasting

Advantages	Disadvantages
The amount of network bandwidth required can be greatly reduced using multicasting. Only one copy of a video stream needs to be sent along any branch in the network.	All viewers of a multicast get the same video at the same time. Individual users cannot pause, rewind or fast-forward the content.
Video sources can be much simpler; they are only required to transmit a single copy of the stream.	The network equipment must be multicast enabled along the entire route from the source to every multicast destination. This can require reconfiguration or possibly hardware/software upgrades for some legacy equipment.
Higher quality (i.e., higher-bandwidth) video can often be used for multicasting.	Some firewalls and NAT devices can block protocols used in multicasting.
	The burden on routers can be significant. They must process multicast control messages and perform packet replication.
	Controlling access to specific video content can be complicated on a multicast network.
	When hybrid public/private networks are used, system installation can be complicated.

5.4.5 Joining and Leaving a Multicast

We already have mentioned before that in multicast all users receive the same video stream at the same time. Users can join or leave a multicast which already started but they cannot start or stop it. They should join the multicast at the time the program happens. In some implementations, users can watch what they have missed (if they are not on time), because these multicast programs are sent in a continuous loop. Multicast sources have to announce the availability of its data stream to the network users (accomplished by SAP packets). Users who are interested in receiving a particular multicast stream have to listen to these announcements, because they contain details on how the multicast is configured. The user devices then take these details and create a join request for the desired multicast. Each router has to track all the multicast streams it is currently processing and if the router is already receiving a stream, it must make a copy for any new users that request it. The bandwidth between routers is used very efficiently, because only one copy of the multicast stream needs to be sent to the routers that have users who are using the stream.

5.5 Application for streaming media

During the last decade, streaming became a primary medium for the delivery of video and audio content. There are several ways, how streaming can be used. First, as standalone for delivering audio and video content and second, as a component in a wider system (so called rich media).Synchronized Multimedia Integration Language (SMIL) gave content producers tools to assemble video, audio and animated vector graphics. This is the kind of multimedia presentation, which can be used for sales promotion, distance learning and product launches.

In the next sections, we will talk more detailed about the following applications for streaming:

- Training
- Entertainment (Internet Radio and Video on Demand)
- Corporate Communications
- Marketing
- Advertising

Of those mentioned applications, some use streaming as primary focus while the others use it as a tool.

5.5.1 Rich Media

In contrast to web content, which was limited to text and still images, it were tools like Quick Time, which allowed to combine text, graphic and video info into an interactive presentation. Later, when Flash was developed, it was used to enrich their static web pages with compelling vector graphics.

Before the development of Flash and other tools, the mix of different formats was restricted to the CD-ROM. Now, because of the advent of streaming media, regular video can be added to web pages and this mix of formats can be delivered real-time as a stream. A very important aspect of a rich media project is, to hold the attention of the audience. There are also some rules people should follow, when they want to create a rich media project:

- Limit the amount of information presented
- Keep graphics simple
- Images should have a good quality
- Make it interesting (e.g. involve the viewer (web chat))

Such a project combines for example streaming media, Power Point slides, text transcripts and interactive facilities like web chat.

5.5.2 Training

A much appreciated application of streaming is training or distance learning. The learner feels the bond of video communication, while he hears and sees the lecture, interacts with forums and chats, but still can be separated in time and space from the lecturer. For background reading, conventional synchronized rich media projects are backed with linked documents. The training can be called up from an archive as a live webcast or a live lecture.

5.5.3 Entertainment

With the advent of DRM and better content networks, content owners are nowadays better prepared to sell content via the Internet. A while ago, early distribution networks had a bad quality. This makes unwilling buyers to look for alternatives like peer to peer networks, which also have the advantage of being for free (although this is illegal).

5.5.3.1 *Internet Radio*

The technology of Internet Radio has reached maturity and the bandwidth demands are reasonable, while the audio quality can be very good. This makes it possible to listen to a vast amount of streamed radio station from all around the world, which cover basically all genres.

5.5.3.2 *Mobile*

Especially since 3G cellular wireless networks are easy available for people, it became feasible to deliver video to mobile devices. Short clips can be streamed or downloaded and subsequently played. There are several codecs (MPEG-4, proprietary codecs) which can be used to encode the content.

A good example for a proprietary codec is Oplay, a lightweight player that can run on Java, Symbian platforms and GPRS networks (minimum bandwidth requirement is 10 kbits/s). A big advantage of cell phones is, that they already posses a system for billing for premium services.

5.5.3.3 *Video on Demand*

Because expensive storage and distribution systems were needed for conventional television to deliver unicast content, VOD was a great challenge to many operators. This is why until recently, VHS and later DVD rental outlets set the benchmark for price and picture quality. But times have changed. Storage and Servers are becoming cheaper and cheaper, and the advent of video compression makes VOD-services cost-effective. Content can be delivered over cable television circuits or broadband phone connections (ADSL). IP set-top boxes allow VOD to be streamed and then displayed on a regular television receiver.

5.5.3.4 *Corporate Communication*

Corporate webcast has become very important and its possibilities are immense. Many departments, especially those with many remote sites are streaming to convey information, replace meetings and communicate to distributors, agents and resellers.

5.5.3.4.1 Staff Training

It can be very expensive for a company to train its staff. Dedicated training centers can be far away which makes the travel costs very high. Companies looked for alternatives and found the CD-ROM; even though popular, the production costs are high, if interactive presentations are used. Rich media is a good and cost-effective alternative. First, the presentation can be prepared offline (e.g. with Microsoft Producer), to this presentation a live presentation can be added (e.g. with Apreso). The live presentation is encoded as video and audio file and the slides are captured as they change. This makes staff training to be a prime user of multimedia technology. Because the live stream is archived, the viewers can watch the presentation during a time that is more convenient for them. VS Webcasting, developed by Virage, is another tool which is frequently used for this purpose. It adds the facility of interactivity with the audience to answer questions, conduct polls and collect viewing statistics.

5.5.3.4.2 The news site

Financial news sites are amongst the most successful applications for streaming, especially because traders have an immense demand for content and they also have broadband access to the Internet. Using rich media, interviews can be combined with text stories, generated from agency feeds.

5.5.3.5 *Marketing*

Streaming is a useful tool in marketing. It can be used here to inform sales personnel and customers about new products.

5.5.3.5.1 Product Launches

Conventional product launches are a live event. But because companies, especially the big ones have remote offices, this is not always possible. This is where Rich Media offers a useful way to share the experience of a product launch with any viewer, who has connections to the Internet.

5.5.3.5.2 Music Promotion

Music was always strongly associated with the web, especially the use of Internet to download MP3 audio files. This is why the web is used to promote and retail music. Because the potential audience (younger people) may not have access to broadband connections, it is useful to create two versions of a web site (one version for broadband and another version for slower connections). The media player is embedded into the background image for stylistic reasons.

5.5.3.5.3 Advertising

As we already know from broadcast television, video is a very suitable format for delivering marketing messages. Brands did not want to deliver information about their new product as a stalling tiny video. But with the rise of broadband usage and since CDNs can offer a higher quality of service, streamed video advertisement are gaining acceptance. It is a well known fact, that viewers dislike pop-up ads and banners. This is why advertisers are looking for different ways for delivering their message. Macromedia´s Flash is very popular for this purpose, because it is almost ubiquitous in Web browsers and it is very easy to upgrade. It would be against the advertisers’ interest to place technical barriers in the way of the potential viewer and their messages. The video can also be streamed to digital signage for live commercials at the point of sale.

6 Chapter 6 Streaming Protocols

6.1 Internet Protocols: TCP, UDP

On top of IP are the end-to-end transport protocols, where Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) are the most important.

For media streaming, TCP is not appropriate because it favors reliability over timeliness, therefore compressed media data is usually transmitted via UDP/IP, despite control information is usually transmitted via TCP/IP.

UDP/IP delivers the packets from the server to the client faster than TCP/IP. UDP uses a simple transmission model without providing reliability, ordering, or data integrity and promotes the continuous stream of data packets. For UDP, delivering packets means preferable to wait for delayed packets, while in a transmission using TCP/IP protocol, the server will re-request a packet in case it gets lost along the way.

In a media streaming transmission, losing a packet is almost imperceptible (unnoticeable).

The loss of a packet during streaming simply adds "surface noise" to an audio stream or degrades the picture quality of a video.

For a media streaming transmission, IETF has specified a number of protocols for its delivery, control and description over the Internet. These are RTSP (Real-Time Streaming Protocol), RTP (Real-time Transport Protocol) and RTCP (Real-time Transport Control Protocol), which run on top of TCP/IP and UDP/IP.

6.2 RTP (Real-Time transport protocol)

RTP was developed by IETF in January 1996. Also included was a copy of RTP specification on the first revision of ITU recommendation H.323 in 1997, which provides a framework, consisting of media transport, call signaling and conference control.

The standard Real-time Transport Protocol (RTP) builds on top of UDP/IP transport and provides timing recovery, loss detection and correction, payload and source identification, reception quality feedback, media synchronization, and membership management to enable the development of robust systems.

RTP is divided in 2 parts: the data transfer protocol (RTP) and an associated control protocol (RTCP). The RTP data transfer protocol manages the delivery of real-time data between end systems, such as audio and video.

It defines an additional level of framing for the media payload, incorporating a sequence number for loss detection, timestamp to enable timing recovery, payload type and source identifiers, and a marker for significant events within the media stream.

The RTP control protocol (RTCP) provides reception quality feedback, participant identification, and synchronization between media streams. RTCP runs alongside RTP and reports this information periodically.

6.2.1 Profiles

RTP protocol specification is deliberately limited in some aspects: It does not specify algorithms for media play out, timing regeneration, synchronization between media streams, error concealment and correction or congestion control.

The reason why these aspects are not specified is because the standard was supposed to use different applications with different needs and therefore a single behavior cannot be specified.

The necessary information for these algorithms to operate after being specified, is provided through RTP profiles and payload formats

For each different application (e.g., audio, video), RTP defines a *profile* and one or more associated *payload formats*.

The *profile* defines the codecs used, to encode the payload data and their mapping to payload format codes in the Payload Type (PT) field of the RTP header.

The RTP profile for Audio and video conferences with minimal control (*RFC 3551*) defines a set of static payload type assignments and a mechanism for mapping between a payload format and a payload type identifier (in header), using *Session Description Protocol* (SDP).

6.2.2 Payload formats

The payload format defines how particular media types are transported within RTP and they may also define certain properties of the RTP data transfer protocol.

The output produced by a media codec is translated into a series of RTP data packets. Some parts are mapped onto the RTP header, some into a payload header and most into the payload data. The complexity of this mapping process depends on the design of the codec and on the degree of error resilience required.

At its simplest, a payload format defines only the mapping between media clock and RTP timestamp, and mandates that each frame of codec output is placed directly into an RTP packet for transport. An example of this is the payload format for G.722.1 audio.

Many payload formats have been defined, matching the diversity of codecs that are in use today, some of the audio payload formats commonly used are: G.711, G.723.1, G.726, G.728, G.729, GSM, QCELP and MP3H. The commonly used video payload formats include H.261, H.263, H.264 and MPEG.

6.2.3 Mixers and Translators

The RTP packet includes a source identifier (SSRC), which identifies the particular sender from the group. But there are two special kinds of sources: a mixer and a translator.

A mixer combines packets from multiple senders and forwards them to one or more destinations. The mixer assigns itself as the sender of the packet and also re-synchronizes the sending (SSRC).

A translator may change the format of the data in the packet, for example if there is a difference in the allowable transfer rate of the end-points.

6.3 RTP Data Transfer Protocol

The data transfer protocol RTP works with the transfer of real-time data. It provides information of timestamps (for synchronization), sequence numbers (for packet loss and reordering detection) and the payload format (encode format).

6.3.1 RTP Sessions

A *session refers to a group of* participants who are communicating, using RTP. Each participant is identified by a network address and port pair to which data should be sent and a port pair on which data is received.

RTP sessions are designed to transport a single type of media; in a multimedia communication, each media type should be carried in a separate RTP session (in our case one for video and one for audio). Each port pair comprises two adjacent ports: An even-numbered port for RTP data packets and the next higher (odd-numbered) port for RTCP control packets.

The sessions can be *unicast* (a point-to-point session) or *multicast (to* a group of participants).

6.3.1.1 *The RTP Data Transfer Packet*

A RTP data transfer packet is divided into 4 parts:

1. The mandatory RTP header

2. An optional header extension

3. An optional payload header (depending on the payload format used)

4. The payload data itself

<table>
<tr><td>V</td><td>P</td><td>X</td><td></td><td>CC</td><td>M</td><td></td><td>PT=201</td><td>Sequence number</td></tr>
<tr><td colspan="9">Timestamp</td></tr>
<tr><td colspan="9">Synchronization source (SSRC)Identifier</td></tr>
<tr><td colspan="9">Contributing source (CSRC) Identifiers (if mixers are used)</td></tr>
<tr><td colspan="9">Header extension (optional)</td></tr>
<tr><td colspan="9">Payload header (payload format dependent)</td></tr>
<tr><td colspan="9">Payload data
Padding</td></tr>
</table>

Figure 9.RTP Data Transfer packet.

V= Version Number
P= Padding
X= Extension
CC=Count of contribution sources
M=Marker
PT=Payload Type

The mandatory RTP data packet header has a typical size of 12 bytes. After the header, an optional header, which can be 4 to 60 bytes of length, may appear. The fields in the mandatory header are the payload type, sequence number, time-stamp and synchronization source identifier. In addition, there is a count of contributing sources, a marker for interesting events, support for padding, a header extension and a version number.

The fields in the header which are mandatory are as follows:

- **X (Extension)**: (1 bit) Indicates presence of an *Extension header* between standard header and payload data.
- **CC (CSRC Count)**: (4 bits) Contains the number of CSRC identifiers (defined below) that follow the fixed header.
- **M (Marker)**: (1 bit) Used at the application level and defined by a profile. If it is set, it means that the current data has some special relevance for the application.
- **PT (Payload Type)**: (7 bits) Indicates the format of the payload and determines its interpretation by the application. This is specified by an RTP profile.
- **Sequence Number**: (16 bits) the sequence number is incremented by one for each RTP data packets sent and is to be used by the receiver to detect packet loss and to restore packet sequence. The RTP does not specify any action on packet loss.
- **Timestamp**: (32 bits) Used to enable the receiver to play back the received samples at appropriate intervals.
- **SSRC**: (32 bits) Synchronization source identifier uniquely identifies the source of a stream. The synchronization sources within the same RTP session will be unique.
- **CSRC**: Contributing source IDs enumerate contributing sources to a stream which has been generated from multiple sources.

6.3.1.1.1 Payload Data

The final part of an RTP packet is formed by one or more frames of media payload data, directly following any payload header. The size and format of the payload data depend on the payload format and format parameters chosen during session setup.

Many payload formats allow for multiple frames of data to be included in each packet. The number of frames can be determined by inspecting the size of the packets, when the frames have a certain size, but also the payload format can include an identifier in each encapsulated frame that indicates the size of the frame.

There is no limit in the number of frames, but receivers are expected to handle reception of packets with a range of sizes: The guidelines in the audio/video profile suggest up to 200 milliseconds worth of audio, in multiples of the frame size, and video codec should handle both fragmented and complete frames.

6.3.1.1.2 Packet Validation

RTP packets do not contain an explicit protocol to validate the packets, but it is possible to observe the progression of header fields over several packets. We can quickly obtain strong confidence in the validity of an RTP stream.

Possible validity checks that can be performed on stream of RTP packets are outlined in Appendix A of the RTP specification. There are two types of tests:

1. Per-packet checking based on fixed known values of the header fields. For example, packets in which the version number is not equal to 2 are invalid, as are those with an unexpected payload type.
2. Per-flow checking based on patterns in the header fields. For example, if the SSRC is constant, the sequence number increments by one with each packet received and the timestamp intervals are appropriate for the payload type. This is almost certainly an RTP flow and not a misdirected stream.

6.4 TP Control Protocol (RTCP)

RTCP’s main function is to send periodically reports of the quality aspects of the media distribution during a session and transmit this data to the session media source and other session participants. Such information may be used by the source for adaptive media encoding (codec) and detection of transmission faults.

RTCP also provides identifiers to all session participants, the CNAME. This identification establishes unique identification of end-points across an application instance.

RTCP must include session bandwidth management. RTCP reports are expected to be sent by all participants. Such traffic will increase proportionally with the number of participants and maybe causes network congestion. Therefore the frequency of report transmission is dynamically controlled. RTCP bandwidth usage should generally not exceed 5% of total session bandwidth. Furthermore, 25% of the RTCP bandwidth should be reserved to media sources at all times, so that in large conferences new participants can receive the CNAME identifiers of the senders without excessive delay.

A fourth, optional feature, is the provisioning of session control functions, because RTCP is a convenient means to reach all session participants, whereas RTP itself is not. RTP is only transmitted by a media source.

6.4.1.1 *RTCP Packets Formats*

All participants in a session should send compound RTCP packets and, in turn, will receive the compound RTCP packets sent by all other participants.

Each RTCP packet starts with a header similar to that of the RTP data packets. The payload type field identifies the type of the packet. There are five RTCP payload types (200-204) defined:

- Receiver Report (RR)
- Sender Report (SR)
- Source Description (SDES)
- Goodbye (BYE)
- Application-defined packet (APP)

The header is the same for all five packet types. Its size is 4 bytes, comprising five fields:

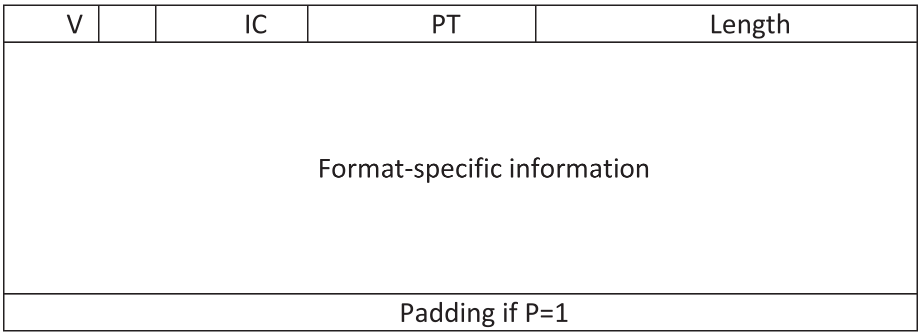


Figure 10. The Basic RTCP Packet Format

V= Version number
P= Padding
IC= item count
PT= packet type

1. **Version number (V)**. The version number is always 2 for the current version of RTP.
2. **Padding (P)**. The padding bit indicates that the packet has been padded out beyond its natural size
3. **Item count (IC)**. Some packet types contain a list of items. The item count field is used by these packet types to indicate the number of items included in the packet (the field has different names in different packet types depending on its use).
4. **Packet type (PT)**. The packet type identifies the type of information carried in the packet. Five standard packet types are defined in the RTP specification.

5. **Length**. The length field denotes the length of the packet contents following the common header.

The RTCP header is followed by the packet data (the format of which depends on the packet type) and optional padding. The combination of header and data is an RTCP packet.

The contents of these packets are in detail described in the following:

6.4.1.1.1 Receiver Report (RR)

One of the main uses of RTCP is quality feedback, which is accomplished through RTCP receiver report (RR) packets, which are sent by all participants who receive data.

<table>
<tr><td>V</td><td></td><td>RC</td><td>PT=201</td><td>Length</td></tr>
<tr><td colspan="5">SSRC of the sender</td></tr>
<tr><td colspan="5">SSRC of the first source</td></tr>
<tr><td colspan="3">Fract. lost</td><td colspan="2">Cum. No of packets lost</td></tr>
<tr><td colspan="5">Ext. highest sequence number received</td></tr>
<tr><td colspan="5">Inter-arrival jitter estimate</td></tr>
<tr><td colspan="5">Last sender report timestamp (LSR)</td></tr>
<tr><td colspan="5">Delay since last sender report (DLSR)</td></tr>
<tr><td colspan="5">…</td></tr>
<tr><td colspan="5">Last reception report block</td></tr>
</table>

Figure 11. Format of the Receiver Report

Each report block describes the reception quality of a single synchronization source from which the reporter has received RTP packets during the current reporting interval. A total of 31 report blocks can be in each RTCP RR packet. If there are more than 31 active senders, the receiver should send multiple RR packets in a compound packet. Each report block has seven fields, for a total of 24 octets. The first 32-bit word in that block is the SSRC of the source, for which this reception report is aimed. The fraction lost field indicates the number of packets lost, divided by the number of packets expected (according to the highest sequence number received) since last receiver report. The lower part of the next 32- bit word includes the highest sequence number received since last report, whereas the higher part is used as an extension to the sequence number revealing possible resets of the sequence numbering. The contents and use of the inter arrival jitter field.

The *inter-arrival jitter* is an estimate of the statistical variance in network transit time for the data packets sent by the reportee synchronization source. Inter arrival jitter is measured in timestamp units.

6.4.1.1.2 Sender Report (SR)

In addition to reception quality reports from receivers, RTCP conveys sender report (SR) packets sent by participants that have recently sent data. These provide information on the media being sent, primarily so that receivers can synchronize multiple media streams (for example, to lip synchronization for audio and video).

V		RC	PT=200	Length
SSRC of the sender				
NTP timestamp (MSB)				
NTP timestamp (LSB)				
RTP timestamp				
Sender's packet count				
Sender's octet count				
First reception report block (SSRC_1)				
…				
Last reception report block (SSRL_n)				

Figure 12. Format of the Sender Report

The first 32 bits of the header of the sender report (Table 12) consists of several control bits. The version number (V) and padding field (P) are the same as in RTP packet. The reception report count (RC) indicates the number of receiver reports attached to this packet that cannot be more than 32. The payload type (PT) for sender report is 200. The length field indicates the length of the packet in 32-bit words minus one. The second 32-bit word includes the SSRC of the sender and the next two words include the high and low parts of the 64-bit NTP (Network Time Protocol) timestamp. The timestamp indicates the time at which this RTCP SR packet was sent. Sender's packet count is the number of data packets that this synchronization source has generated since the beginning of the session. The sender's octet count is the number of octets contained in the payload of those data packets (not including the headers or any padding).

6.4.1.1.3 Source Description (SDES)

This packet provides participant identification and supplementary details, such as location, e-mail address and telephone number. The information in SDES packets is typically entered by the user and is often displayed in the graphical user interface of an application.

<table>
<tr><td>V</td><td></td><td>SC</td><td colspan="2">PT=202</td><td colspan="2">Length</td></tr>
<tr><td colspan="7">SSRC/CSRC of the sender</td></tr>
<tr><td colspan="3">Type</td><td colspan="2">length</td><td colspan="2">text</td></tr>
<tr><td colspan="7">Text continued</td></tr>
<tr><td colspan="7">…</td></tr>
<tr><td colspan="7">Last chunk</td></tr>
</table>

Figure 13.Format of the source description

Each source description packet uses RTCP packet type 202. SDES packets comprise zero or more lists of SDES items, the exact number denoted by the SC header field, each of which contains information on a single identified source.

Each SDES item starts with an 8-bit type field, followed by an 8-bit octet count, which identifies the length of the following text field. The defined SDES items are: Canonical end-point identifier (CNAME), which should follow the format user@host, user name (NAME), being the real user name, electronic mail address (EMAIL) in format name@corporation.com, phone number (PHONE), geographical user location (LOC), application or tool name (TOOL), notice (NOTE) and private extensions (PRIV). Only the item CNAME is mandatory.

6.4.1.1.4 Goodbye (BYE) packet

<table>
<tr><td>V</td><td></td><td>SC</td><td>PT=203</td><td>Length</td></tr>
<tr><td colspan="5">SSRC/CSRC of the sender</td></tr>
<tr><td colspan="3">length</td><td colspan="2">reason for leaving</td></tr>
<tr><td colspan="5">…</td></tr>
<tr><td colspan="5">Last chunk</td></tr>
</table>

Figure 14. Format of the BYE packet

A BYE packet may also contain text, indicating the reason for leaving a session, suitable for display in the user interface. This text is optional, but an implementation must be prepared to receive it (even though the text may be ignored).

6.4.1.1.5 Application-defined packet (APP)

Application-defined packets are used for nonstandard extensions to RTCP and for experimentation with new features. The intent is that experimenters use APP as a first place to try new features and then register new packet types if the features have wider use. Several applications generate APP packets and implementations should be repaired to ignore unrecognized APP packets.

V	P	Sub	PT=204	Length
SSRC/CSRC of the sender				
Name (ASCII)				
Application-dependent data				

Figure 15.Format of the application defined packet

6.4.1.2 *Packet Validation*

It is important to validate whether received packets really are RTP or RTCP. The packing rules, mentioned earlier, allow RTCP packets to be rigorously validated.

6.5 RTSP. Real Time Streaming Protocol

RTSP was developed by the Multiparty Multimedia Session Control Working Group (MMUSIC WG) of the (IETF) and published as RFC 2326 in 1998.

It is a client-server multimedia protocol that establishes and controls one or a number of streamed multimedia data over IP network. RTSP provides "VCR-style" remote control functionality for audio and video streams, like pause, fast forward, reverse, and absolute positioning, to make real -time control of playback of media files from the server possible. RTSP is an application-level protocol designed to work with lower-level protocols like RTP, RSVP to provide a complete streaming service over internet.

It uses TCP for data control of the reproduction and UDP for audio and video data. It works for large audience multicast, as well as single-viewer unicast.

RTSP objective is to provide the same services on streamed audio and video just as HTTP does for text and graphics, therefore it has similar syntax and operations so that most extension mechanisms to HTTP can be added to RTSP.

Each presentation and media stream is identified by an RTSP URL. The overall presentation and the properties of the media are defined in a presentation description file, which may include the encoding, language, RTSP URLs, destination address, port and other parameters. The presentation description file can be obtained by the client using HTTP, email or other means.

6.5.1 Operations

RTSP is the "network remote control" between the server and the client. It provides the following operations:

- Recovery of media from media server: The client can request a presentation description and ask the server to setup a session to send the requested data.
- Invitation of a media server to a conference: The media server can be invited to the conference to play back media or to record a presentation.
- Adding media to an existing presentation: The server or the client can notify each other about any additional media becoming available.

6.5.2 Methods

The media server and the client can issue requests (methods) in RTSP. These requests are usually sent on a channel independent of the data channel. They can be transmitted in persistent transport connections or as a one connection per request/response transaction or in connectionless mode.

The services and operations are supported through the following methods:

- **OPTIONS:** Return the request types the server will accept.
- **DESCRIBE**: The client retrieves the description of a media object identified by the request URL from the server.
- **ANNOUNCE**: When sent from client to server, it posts the description of a media object, identified by the request URL to a server. When sent from server to client, it updates the session description in real-time.
- **SETUP**: The client asks the server to allocate resources for a stream and start an RTSP session.
- **PLAY**: The client asks the server to start sending data on a stream allocated via SETUP.
- **PAUSE**: The client temporarily halts the stream delivery without freeing server resources.
- **TEARDOWN**: The client asks the server to stop delivery of the specified stream and free the resources associated with it.
- **GET_PARAMETER**: Retrieves the value of a parameter of a presentation or a stream specified in the URI.
- **SET_PARAMETER**: Sets the value of a parameter for a presentation or stream specified by the URI.
- **REDIRECT**: The server informs the clients that it must connect to another server location. The mandatory location header indicates the URL the client should connect to.
- **RECORD**: The client initiates the recording of a range of media data according to the presentation description .The server decides whether to store the recorded data under the request URI or another URI.

7 Chapter 7

Inserting Video into Websites

With the website development tools available today, it is possible to include many different types of content in web pages, such as photographs, illustrations, sound clips, animated buttons and video content. With the proper tools, HTML or XML code can be written to format the web pages, and any media files encountered by browsers are handled by plug-ins for user playback. So, in a basic sense, all that is needed to embed video in a website is a suitable tool.

7.1 HTML

Hypertext Markup Language, or HTML, is the main programming language used to create web pages.

An HTML file is a basic text file with the filename extension "HTML" or "HTM", which employs "markup tags" to indicate the text format. The written form of HTML elements consists of tags enclosed in angle brackets (<tag name>). HTML tags most commonly come in pairs like <h1> and </h1>, although some tags, known as empty elements, are unpaired, for example <img>. The first tag in a pair is the start tag; the second tag is the end tag. In between these tags, web designers can add text, tags, comments and other types of text-based content.

HTML elements form the building blocks of all websites. HTML allows images and objects to be embedded and can be used to create interactive forms. It provides means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. It can embed scripts in languages such as JavaScript, which affect the behavior of HTML web pages.

A typical HTML file is stored on a Web server where it is accessed by a client using a Web browser. The purpose of a web browser is to read HTML documents and build them into visible and audible web pages. It does not display the HTML tags; it only uses them to interpret the content of the page. The Web browser acts as a container for texts and "objects" such as images and sounds. The objects are actually not stored in the document. Instead, tags containing external reference to objects are inserted at appropriate places in the HTML text. Thus, an HTML page actually consists of the HTML file itself, along with any additional references, graphics and multimedia files. All of these objects must be online and available to a user when the file is opened.

Web browsers can also refer to Cascading Style Sheets (CSS) to define the appearance and layout of text and other material. The W3C, maintainer of both the HTML and the CSS standards, encourages the use of CSS over explicitly presentational HTML markup.

7.1.1 A Short Introduction to HTML

7.1.1.1 *Tags and attributes*

HTML defines 91 tags that designers can use to mark the different elements of a page. In spite of the fact that this is a very large number of labels, it is not enough to create complex pages. Some elements such as images and links require some additional information to be fully defined. Although each of the HMTL tags defines its own attributes, some attributes are common to many or almost all tags. The common attributes are divided into four groups according to their functionality:

Basic attributes: It can be used in almost all HTML tags. Some of these are *id, class, style* or *title*.

1) Internationalization attributes: Those are used by pages that show their contents in multiple languages or those who want to indicate explicitly the language of their contents, for example *lang* and *xml:lang*.
2) Event attributes: Those are only used in dynamic web pages created with JavaScript. Some examples are *onclick, ondblclick, onmousedown, onmouseup, onmouseover, onmousemove, onmouseout, onkeypress, onkeydown, onkeyup*.
3) Attributes for elements that can be focused: When the user selects an element from the interface of an application, it is said that "the element has the focus of the program". The web page elements can focus on the application) (in this case, the focus of the browser) and HTML defines some specific attributes like *accesskey, tabindex, onfocus*y*Onblur* to control how the elements are selected.

7.1.2 Elements

In addition to tags and attributes, HTML defines the term element to refer to HTML parts that form the documents. HTML elements are composed of: An opening tag, zero or more attributes, text enclosed by the tag and a closing tag. Some of the most important elements are the following:

- Tables: <table> is used to define data tables, <tr> is used to define each row in the table and <td> is used to define each column of the table.
- Images and Objects: <img> is used to include images in documents
- Forms: <form>is used to insert a form on the page and <input> is used to insert a control on the form.
- Links: <a> is used to link all types of resources and <link> is used to link and set relationships between the document and other resources to link a file.
- Lists: <ul> is used to define an unordered list, <ol> is used to define an ordered list and <li> is used to define each element of the list.

Below is shown an example of the structure from a HTML element, the paragraph. It begins with an opening tag (<p>), contains zero or more attributes (class="normal"), has a text content (This is a paragraph) and ends with a closing tag (</p>).

<p class=”normal”>This is a paragraph</p>

Moreover, the HTML language classifies all the elements in two groups: Inline elements and block elements. The main difference between the two types of elements is the way they occupy space on the page. Block elements always start on a new line and occupy all available space until the end of the line although the contents don´t reach the end of the line. The inline elements occupy only the space necessary to display its contents.

7.1.3 Data types

HTML defines several data types for element content, such as script data and style sheet data, and a plethora of types for attribute values, including IDs, names, URIs, lists, tables, forms, numbers, units of length, languages, media descriptors, colors, character encodings, dates and times, and so on. All of these data types are specializations of character data.

7.2 HTML y CSS

Originally, only HTML pages included information about their contents and text images. With the development of the HTML standard, the pages began to include also information about the appearance of their contents: Fonts, colors and margins.

The subsequent emergence of technologies such as JavaScript, prompted to HTML pages to include the application codes (called scripts) that are used to create dynamic web pages.

The design and the programming, both of which are included in the same HTML page as the content, complicate excessively its maintenance. Normally, the content and layout of the website are the responsibility of different people, so it is appropriate to separate them.

CSS is the mechanism that allows separating the contents defined by HTML from the appearance of the contents.

Another advantage of the separation of content and presentation is that the created HTML documents are more flexible and this is why they are better suited for different platforms: Computer screens, mobile screens, printers and devices used by disabled people.

7.3 HTML5

HTML5 is the fifth major revision of the standard language of the World Wide Web, HTML. It is going to bring a series of important improvements, as well as new markup tags while other tags will disappear. As a result, the code is simplified.

One of the points to highlight in HTML5 is that it offers more power at the time of web development, without having to switch using other technologies. Currently for example, platforms like Apple's iOS and Google´s Android require developers to create unique codes for each platform so that the application can communicate with the platform, and vice versa. HTML5, on the other hand, is a universal language, meaning that developers only have to create one codebase.

From the W3C Web, we found an introduction to the new improvements of HTML5.

"Some of the most interesting new features for authors are APIs for drawing two-dimensional graphics, embedding and controlling audio and video content, maintaining persistent client-side data storage and to offer users editing documents, or parts thereof, interactivity. Other features facilitate the representation of familiar page elements, including <section> (section) <footer> (foot); <nav> (for navigation) and <figure> (for assigning a caption to a photo or other content contained on page). Authors write HTML5 using HTML syntax "classic" or XML syntax, according to the needs of the application."

7.3.1 New HTML5 tags

Most webs have a structure of type header - menu - body – footer. With/In HTML5, specific tags are used to structure the site, instead of having to create div for each part of the website.

• <header>

• <nav>

• <section>

• <figure> assigning a caption to a photo

7.3.1.1 *New tags for video and audio*

Tags like <embed> and <object> are forgotten when it comes to include audio, flash animations or video. Hence, with the new version of HTML5 come two new tags <video> and <audio>.<canvas> is another tag that will allow us to work with graphics and animations.

7.3.1.1.1 Tag <video>

First of all, the video tag allows you to add your videos directly to your pages without requiring any plug-in based solution, like Flash. Thanks to the video tag, you also don't need any plug in to view videos on most mobile devices and computers.

Major browser support for the video tag is limited to Mozilla Firefox 3.5 or higher and Google Chrome 3.0 or higher. Internet Explorer 8 does not support the video tag (that is, without the use of Google Chrome Frame), though it will reportedly be supported in the upcoming Internet Explorer 9.

As an example, the following HTML code will embed a video with standard playback controls built into the video element:

Listing 1.The Tag video

```
<video src="video.ogv" controls>
        Content
</video>
```

The *src* attribute specifies the video file to play, and the *controls* attribute indicates that the browser should provide its own playback controls. Any content inside the video tag will be rendered in browsers that do not support HTML5 video.

Using JavaScript to Control HTML5 Video

Many features of the video tag can be automated using JavaScript. The following code demonstrates how to use JavaScript to play, pause, and seek the video.

Listing 2.Using JavaScript with the Tag video

```
<video src="video.ogv" id="v">
</video>

<p>
<button onclick="document.getElementById('v').pause()">
  Pause</button>
<button onclick="document.getElementById('v').play()">
  Play</button>
<button onclick="document.getElementById('v').currentTime = 0">
  Back to Beginning</button>
<button onclick="document.getElementById('v').currentTime += 5">
  Skip 5 Seconds</button>
</p>
```

The *pause*() and *play*() methods are self-explanatory. The *currentTime* property returns the current playback location in seconds. Setting the *currentTime* property will result in the video seeking to the specified location.

7.3.1.1.2 Tag<audio>

The <audio> tag defines sound, such as music or other audio streams. It is supported in Internet Explorer 9, Firefox, Opera, Chrome, and Safari. Also, it is simpler to use than the video tag, because it only involves an audio player. It is very easy to implement:

Listing 3.The Tag Audio

```
<audio src="/test/audio.ogg">
```

</audio>

The *src* attribute can be a URL of the audio file or the path to the file on the local system. In the following, we will explain the attributes of the audio tag:

Listing 4.Properties of the Tag Audio

```
<audio src="audio.mp3" preload="auto" controls></audio>
```

The *controls* display the standard HTML5 controls for the audio on the web page. The *autoplay* makes the audio play automatically. The *loop* makes the audio repeat (loop) automatically. And the `preload` attribute is used in the audio element for buffering large files. It can take one out of three values: `"none", which` does not buffer the file, `"auto"`, which buffers the media file and `"metadata", which` buffers only the metadata for/of the file.

Layout engine

Because we have many browsers with different compatibilities, we need rendering engines. These are responsible for rendering the code of our website and then implement multimedia content. These are the engines that correspond to each browser:

- Google Chrome: WebKit
- Safari: WebKit
- Mozilla FireFox: Gecko
- Internet Explorer: Trident
- Opera: Presto

It is a known fact that engines or browsers accept a format and we are going to make all of them to support <audio>. You have to associate the different files that exist with any of the browsers that are able to play these files.

Listing 5.How to associate the file with the tag Audio

```
<audio controls autoplay preload>
<source src="cancion.ogg" type="audio/ogg" />
<source src="cancion.mp3" type="audio/mpeg" />
<source src="cancion.wav" type="audio/wav" />
</audio>
```

It is almost the same, *but now we will remove src*. The other properties are left as they are, but we will put a tag inside the source opening tag and the source closing tag.

- *src*: Links the audio file you want to play.
- *type*: Says what audio type or codec will play.

With this, we support all rendering engines or browsers (except IE8 and earlier, IE9 is supported).

7.3.1.1.3 Tag<canvas>

The canvas tag is used in HTML 5 to create 2D and 3D graphics, and can even interact with graphic elements defined within the canvas.

Listing 6. The Tag Canvas

```
<canvas id="nombre" width="valor" height="valor">

</canvas>
```

It has only three possible attributes, ID, width and height. ID identifies the tag, the other two are optional, and its default size is300x150. The tag creates a "canvas", a particular space where we can draw pixel by pixel using JavaScript.

7.3.2 New APIs

The new language will be much more powerful, because it incorporates several new APIs that will allow us to work with databases locally, interact by the user with different elements of the interface, drag & drop and improve geolocation.

Summing up a bit, the HTML5 brings more power, more performance, less code, and more simplification. It brings major improvements, Websites will load faster and there will be a higher capacity for the interaction by the user. It is a language that is closer/close to Web 2.0 and it maybe will be the next Web 3.0.

Now, we just have to wait to solve compatibility issues with the different browsers, revise the language by W3C, and talk about the new language as “The Future of Web Content”.

8 Chapter 8
Multi- View Scenario

8.1 Introduction

Thanks to continuously changing technologies and the resulting reduction of the cost of the different materials allowed new developments to occur in video technologies. Nowadays, its future is heading towards Multi-view systems. The traditional video is a two-dimensional (2D) medium and only provides a passive way for viewers to observe the scene. However, the multi-view video can offer arbitrary viewpoints of dynamic scenes and thus provides more realistic video to users. This technology leads to a new diversity of applications (as virtual view synthesis, high-performance imaging, image/video segmentation, object tracking/recognition, environmental surveillance, remote education, industrial inspection, 3DTV, and Free Viewpoint TV (FTV)) and research.

Between the new upturns that Multi-view videos offer, the following should be mentioned: Provides better image quality, better estimation of the image, space diversity, more efficient compression, better segmentation and a higher recognition probability.

These new technologies of Multi-view video have as final result one single image which results from a set of image overlays. But there is also the possibility of having another final result, where different video signals can be monitored by a Multi-view display.

In this chapter, the concept, types, and applications of the Multi-view technology will be explained. Additionally, a new scenario of Multi-view will be explained. This scenario is based on different displays, which means that different video perspectives can be monitored at the same time. Its possible distribution will be analyzed, showing a business case, a profitability plan and its costs. The production of the Multi-view scenario will be explained in Chapter 10.

8.2 What is Multi-view Video

Multi-view technology refers to the recording of a scene in different positions; it means that multiple cameras, a minimum of two are placed around the scene at different angles. The viewpoint and the view direction can be changed interactively, allowing the viewer to experiment with free viewpoint navigation, within the range covered by the shooting cameras.

This property of free viewpoint navigation opens the door to multiple possibilities, using the Multi-view technology, these are the following:

- Entertainment : Concert, sports, multi-user game, movie, drama, news
- Education: Cultural archives, manual with real video instruction of sports playing, medical surgery
- Sightseeing : Zoo, aquarium, botanical garden, museum
- Surveillance: Traffic intersection, underground parking, bank
- Archive: Space archive, living national treasures, traditional entertainment
- Art/Content: Creation of new type of media art and digital content

Thanks to this technology, some improvements have been obtained:

- Estimation of illumination, which provides better image quality
- Better image restoration has been reached, thanks to a better estimation of the image
- Spatial resolution (super-resolution) due to the fact that multi-view provides space diversity.
- More efficient compression because of the spatial correlation
- Better segmentation because there is more information about the scene and objects available.

8.3 Types of Multi-view

Multi-view does not only refer to different positions in space, but also to different time lapses and others things. The different types are summarized below.

- Multiple views from different positions or angles (space): It refers to images captured by multiple cameras at the same time. The most known applications are 3DTV, FTV and immersive teleconference.

Figure 16. Example of multiple views from different angles

•Multiple views from different time instants: Such as satellite images captured at different times.

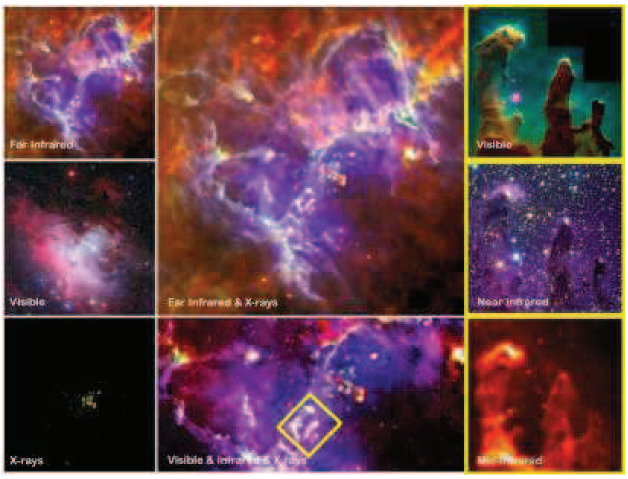


Figure 17. Example of multiple views from different times

•Multiple views from different imaging modalities: For example CT (Computer Tomography), MRI (Magnet Resonance Imaging), Acoustic images, etc.

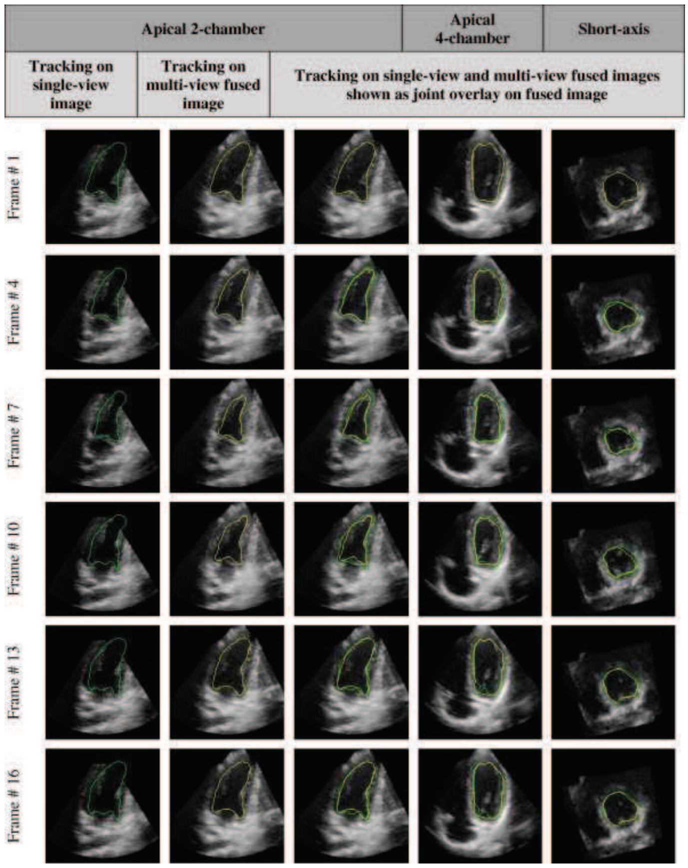


Figure 18. Multiple views from different imaging modalities

In this project, we are going to focus on the first type, the Multi-view from different angles. The following applications explained are also focused on this type.

8.4 Applications

One of the main objectives of the multi-view is to get a realistic visual system. The multi-view video system can provide an augmented realism through selective viewing experience. It is achieved through 3D, where a collection of multiple videos, captured at different viewpoints is obtained.

In the following, three of the more common applications will be explained: FTV, 3DTV and immersive Teleconference.

8.4.1 3DTV

3D television (3DTV) is television that conveys depth perception to the viewer by employing techniques such as stereoscopic displays, multi-view display, 2D-plus-depth, or any other form of 3D display. Most modern 3D television sets use an active shutter 3D system or not polarized 3D systems, while some of them are autostereoscopic, which means that special glasses are not needed.

8.4.1.1 *Stereoscopic 3D*

Any process that allows to create 3D images are known as stereoscopic, and is based primarily on the natural principle of human vision, where each one of our eyes captures in the same instant an image slightly different to the image of the other eye, due to the distance that separates them.

For a correct image, the viewer requires suitable lenses for the combination of both images. It is noteworthy that the point in the space where the viewer is located is not relevant, because the displayed perspective is the same from any angle.

Such 3D display systems have gained significance in niche markets and have already become practical for various applications, such as 3D cinemas (e.g. IMAX theatres), PC gaming, professional and scientific visualization, etc.

8.4.1.2 *Autostereoscopic TV*

The autostereoscopic television is considered an improvement over the previous system and allows viewing 3D TV without glasses. Besides representing the depth information, it allows the arbitrary selection of the viewpoint and direction within the scene. Thus, a change of the position of the viewer affects the observed image. The feeling is that the scene rotates with the movements of the observer. This phenomenon is known as Free Viewpoint. The main application of this technique is called Free Viewpoint Television (FTV).

The difference in capturing the scene in each case is important to note. For the stereoscopic TV, at least two cameras are required to form the complete picture. But for autoestereoscopic TV, it depends on the number of viewpoints that are taken. For example, at least six cameras would be needed for three viewpoints and at least eight cameras would be needed for four viewpoints, etc. Nowadays, the number of viewpoints is limited to eight, due to technological issues.

8.4.2 FTV

The Free Viewpoint Television (FTV) is a 3D information system that includes acquisition, processing and playback.

FTV allows seeing the real world (3D) by just changing the viewpoint as if we were in the scene. It is a system in which the user chooses what they want to see in a scene.

A scene is captured by dozens of cameras connected to a server. On the viewer's side, the user has a monitor in front of him/her, where the images acquired can be observed, and a camera that will detect the movements of the user. The information obtained from the camera about the user´s movement is sent to the server which will select the image to be displayed on the screen. This technology is called window effect. If the user moves to the right, he/she can visualize what is on the left side of the scene and vice versa. This way, FTV not only generates images from a free view point but it also carries spatial information.

8.4.3 Immersive Teleconference

In immersive teleconference, there is an interaction between viewers. Participants at different geographical sites meet virtually and see one another in either free viewpoint or 3DTV style. The immersiveness provides a more natural way of communications.

8.5 New Scenario Concept

A new Multi-view video scenario is presented in this chapter, which has as final result multiple video sequences synchronized between them.

This new Multi-view scenario is formed by a main screen, on which the desired content can be watched, and a side bar, which consists of five selectable small screens. The content of any of these screens can be monitored by prior selection on the main screen. As a new feature, it provides a language selecting tool.

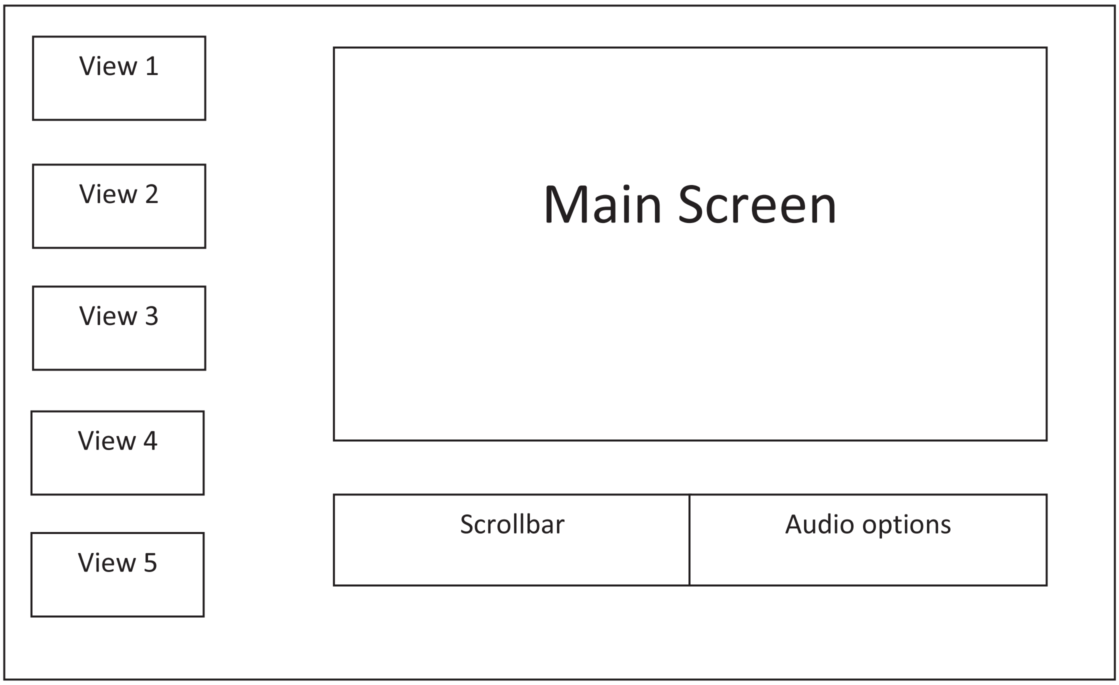


Figure 19. Model of Multi-view video scenario

8.5.1 Synchronization

This different concept of Multi-view can be generated by a single camera or multiple cameras. In this case, a maximum of five cameras is possible. The audio can be recorded in one channel or in different channels with special devices. When the recording process is done with one or two cameras, the synchronization of the videos and the audio is carried out after the recording, in the post-production stage. Certain effects, such as openings, credits, effects or subtitles are also added in this phase.

8.5.2 Interaction

With the New Multi-video scenario, the viewer has the possibility to interact with the system. Using the side-bar area, where the five videos are located, the user can arrange the assignments of the main screen by one click, changing the perspective of the scene.

With the new feature, the interaction with the audio is also possible. The language of visualization can be selected on the audio tool bar, located below the main screen.

8.6 Delivering content

The content can be delivered in multiple forms. The most desirable option is, to stream the Multi-view scenario over IP networks.

8.6.1 Over IP Network

The Multi-view scenario, the audio and video files have to be hosted on a streaming or ftp server. In first instance, it can be imagined that the main problem could be the big bandwidth, needed for delivering five video streams. But thanks to the new codecs available, such as WebM, the videos are compressed to a minimum size and therefore less bandwidth is required.

8.6.1.1 *Live Multi-view*

Because there is no editing process in this case, five cameras are needed in order to broadcast a live Multi-view. The content is codified and streamed out by specific software. In first instance, this type of Multi-view is focused on a private market, such as Sports and Culture.

8.7 Distribution

8.7.1 Overview of multimedia content

What the society demands today is media capable of interactivity and connectivity. These are the so-called emerging media and are possible thanks to technological innovations that tend to integrate a micro universe of multiple digital devices in wireless mobile networks (Bluetooth, Wi-Fi, wi-max) making the user a terminal of permanent connection to multimedia content.

After the overview mentioned above, can be concluded that the future of the media can be found on the web. The data distribution phase is performed using streaming technologies or peering exchanged based on a marketing model "download" and use "on demand".

For getting an adequate positioning between the media data networks such as Internet, intranets, etc that use as main tool support or hypertext the email, web, chat, etc; and contents and/or learning units online as training materials (images, audio, video, documents and to complex multimedia productions.

8.7.2 E-learning

With regard to e-learning, social networks, blogs, wikis, podcasts, labeling systems and social filtering, social bookmarking..., are also elements that have a pedagogical and didactic use and contribute to collaborative learning, the backbone of this new trend in the e-learning world.

The new perspectives offered by these tools complete the LMS systems used in virtual education (on-line), favoring the exchange. Additionally, can be developed specific applications that exploit and adapt the educational material to the technical specifications of the various devices supported. These applications will use the web services available on the platform to request and obtain the material which will be adapted to the specifications of the device to present the activities in a natural way to each display and interaction system.

The need for these individual applications is especially necessary in mobile devices such as mobile with access to the network, which in most cases does not exhibit acceptable compatibility with standard of video broadcast in the network, and any case, do not provide to the user a experience similar to that is achieved in the computer. Thus, we believe that the adaptation not only of the contents, but the interface and interaction on mobile devices is essential in a ubiquitous system as we propose.

8.7.3 HTML5 input

The following explains why HTML5 has a crucial importance on video distribution. The main reason is that HTML5 improves the search engines and provides greater accessibility. The search done by the browser leads to have the expected form the association in the different results of the web pages.

8.7.3.1 *Search engine optimization*

Search engine optimization (SEO) is the process of improving the visibility of a website or a web page in a search engine.

In general, the earlier (or higher ranked on the search results page), and the more frequently a site appears in a search results list, the more visitors it will receive from the search engine's users. SEO may target different kinds of search, including image search, local search, video search, academic search, etc.

The most optimal and current, if not to say the only Internet marketing strategy is called SEO, which is being consider in the search engines work, what people search for, the actual search terms or keywords typed into search engines and which search engines are preferred by their targeted audience.

One of the crucial SEO benefits with HTML5 is being able to add a “Title” attribute to any tag of HTML5, as well as the video tag. The title of your video works just like an h1 tag and can be read by search engines. Additionally, clear and concise browser coding helps search engines to identify and index your video easily.

Using the video tag, content providers (for example YouTube) can have their video content indexed and ranked by the search engines easily, and their content can be viewed by anyone using a current browser or most mobile devices. Thus, the potential viewership of these video content providers increases thanks to the SEO properties of the video tag.

Another great benefit for SEO is that it to allow the video file to receive back links whenever it is shared. In other words, other sites can have a video tag pointing to the video file located on your site. Consequently, when people share the video on their sites, your video receives back links which helps rankings.

Optimizing a website may involve editing its content and HTML5 and associated coding to both increase its relevance to specific keywords and to remove barriers to the indexing activities of search engines. Promoting a site to increase the number of back links, or inbound links, is another SEO tactic.

All these features make the distribution of our videos faster, easier and more efficient.

8.8 Consumption

8.8.1 Overview of multimedia content

The circulation and exchange of multimedia content through video dynamic platforms (YouTube), MP3 audio (Playlist), text documents (Scribd), images (Flickr), office files (Slide Share), digital discography (Pure Volume, ZonaIndi), on-line game platforms and MMS for mobile, are some of the resources available on the WEB and P2P networks.

The following examples present the most common emerging media which consumes the multimedia content in Web 2.0, also called social web.

- **Blogosphere:** A blog is a regularly updated website that collects chronological texts, videos or articles from one or more authors, appearing first the most recent uploaded website, allowing the freedom to go back or to leave according the users will.
- **Podcasting:** The podcasting consist on creating audio files (usually in MP3 or AAC, and in some cases OGG) and video files (called video casts or vodcasts). Its distribution is carried through a RSS file which allows subscription. A program is used to download the podcast from the Internet. In this way the user can listen to it in anytime he wants.
- **Wikis**: A wiki is a website whose web pages can be edited by multiple volunteers through the web browser. Users can create, modify or delete the same text they share (Wikipedia).
- **Social Networks**: Networks in which structure the individual nodes are people who maintain relationships, such as friendship, common or commercial interests (MySpace, Facebook).
- **On-line applications**: satellite maps (Google Maps), virtual disks, image and video editing (BubbleShare, eyespot), shared documents (Google Docs), etc.
- **File sharing platforms**: web publishing environments of multimedia content available for free download (YouTube, Flickr, Google Video, Scribd).

Due to technological behavior in which we live, the demand for these tools is growing. The connectivity and the content personalization have become part of our lifestyle.

8.8.2 E-learning

Thanks to new information and communication technologies (ICT), students 'online' can communicate and collaborate with their peers "class" and teachers synchronously or asynchronously, without time-space limitations.

Now are commented the major distribution platforms e-learning.

In an E- Learning platform a content management system (CMS) is a program that allows creating a support structure for the creation and management of content mainly by participants in web pages. The hardware and software environment designed to automate and manage the development of training activities is known as e-learning platform or learning management system (LMS).

A learning management system (LMS) registers users, organizes course catalogs, stores user data and provides reports to management. Typically included are communication tools to be used by the course participants

. Improvements in usability (easy and intuitive navigation) and accessibility (possibility for people with disabilities) allow to reduce the digital separation and to extend training opportunities to more people, overcoming one of the biggest barriers to e-learning: Technological barrier.

Currently there are a great number of platforms, both commercial and open source. At the university field, the most used platform is WebCT, followed at some distance from the Edustan Ce platform and implement Moodle´s platform (free license). LRN (dot learn), Docebo, Blackboard and eCollege, as well.

8.9 Business Analysis

8.9.1 Executive Summary

Media Streaming refers to the process of electronically distributing digital content over a network. The media streaming process sends audio and video media which is stored on a server to a specified location at a specified time. This type of electronic delivery is referred to as synchronous streaming. Users may view the media files as soon as the data is delivered from the streaming server.

Streaming video can be viewed by anyone, anytime, anywhere; today's technology can track all the details. By getting to know your audience, you can develop targeted campaigns that establish a bond with consumers. For example, educational and training opportunities are not confined to classrooms or companies, since they can simultaneously train countless employees, students; around the world.

Multi-view video Streaming isn't just for luxury brands, entertainment, or news industries. It has numerous common business applications, including company meetings, distance learning, sales force training, surveillance, video email, product introduction, event broadcasts, news distribution, webcasting and web conferencing. Insight Research, reports that streaming video and music will grow at a rate of 29% and generate $70 billion over the next six years.

Simply anything that can become a part of the footage can be kept on a web server and published on the Internet, including entertainment videos, business meetings, political speeches, security monitoring or 'virtual tours' which let the users see the layout of a particular building or virtual education. Live video streaming can also help monitor remote locations.

8.9.2 Mission Statement

The mission of the mul is to offer a multi- view video scenario platform with 5 screens playing with media streaming in a simultaneous way with the possibility of audio selection in different languages and the option to have the subtitles in the original language placed in one of the screens.

8.9.3 Situation Analysis

Media streaming is perfect for corporations who make use of the Internet to attract investors and clients. Since there is now the easier way to create and upload interactive and lively multimedia presentations and other content in their Web sites, they can attract more people to try their services and products. With just the use of digital video cameras, web cameras, music players, and other gadgets, they can easily upload these into their servers whether by using their own equipment, even when they are simple ones, by accommodating streaming requirements. In any of the cases, it is also possible to hire managed services to be sure of the technical expertise of the staff that will monitor the media streaming operations.

The use of Media Streaming together with Multi-view video scenarios provides to the final customer the possibility of a new and interesting platform usable and friendly. The display mode of the platform is convenient for the users, since there is a main screen and then in a side bar the 4 other scenarios are shown, but all of them are playing at the same time regarding their one perspective or setting. All this characteristics, together with the three language option of selection and the leeway for subtitles in the original language in one of the screens made out of this Multi video updated application, a total innovation in the world of streaming and multi video.

8.9.4 Goals

- Streaming offers the possibility for users to take advantage of interactive applications like video search and personalized playlists.

- Allows content deliverers to monitor what visitors are watching and how long they are watching it.

- Provides an efficient use of bandwidth since only part of the file that's being transferred is the part that’s being watched.

- Give out the content creator with more control over his intellectual property because the video file is not stored on the viewer's computer. Once the video data is played, it is discarded by the media player avoiding piracy and keeps the security of the company’s web sites.

- The visitors get the possibility of having different perspectives from a same scenario in a real time after buffering in a synchronized and simultaneous way.

8.9.5 Marketing Strategies

"We aren't selling toasters; we are selling exciting products," David Pryor says in an interview with Automotive News. He's the Vice President of Marketing, and his exciting products are Porsches. "It's very hard to communicate that emotion with just text and pictures."

Businesses from every industry with the need for communication are quickly recognizing the value of streaming media, particularly streaming video, imagine what can a Multi video streaming scenario can offer to visitors, consumers, users, or followers, no matter how you want to define them or circumstance under which they are. A great market can be acquire with a this easy, simple and interesting platform, that is easy to use and can be easily associated to DVD home watch use setting properties.

Nonetheless, is important to emphasize that in order to have an adequate attention caption of customers and therefore the frequency of visits in your Media Streaming, ensuring the success proper planning must begin before the camera starts rolling. Like traditional video, the quality of streaming video products reflects the skill and equipment employed in production.

Video production specialists know how to capture shots that compress well and translate smoothly when streamed in even the smallest window. Any excessive camera motion techniques, including fast cuts, pans and zooms, reduce the speed and quality of streaming video. Tripods, image stabilization, close-ups and fine-tuned encoding can reduce complications in all connection speeds.

Color and contrast also affect compression. Dark colors can be blended with shadows, and patterns must be refreshed at even the smallest movement. Consequently, solid bright colors and subjects that contrast with their backgrounds allow for optimal video quality.

The running time of streaming video is crucial to its effectiveness. A video designed for distribution or displayed at a tradeshow, for example, is probably too lengthy to be streamed from the web. Such a video should either be re-edited into a shorter version, or split into a series of clips.

The adequate running and same time buffering of the 5 video/scenarios in a simultaneous way, as well as the arrangement of all the screens in the display will generate all together the success and viewers acceptance and interest in order to watch it. So must be said that once more in the technical part, though the Multi-view video and different scenarios might be attractive by themselves, quality of the streaming videos in each mini screen will hold in it the platform.

8.9.6 Marketing Communications Tactics

Selecting search engine optimization for promoting your business on the internet, one must knows the ultimate benefits of SEO (Search Engine Optimization) campaign.

Select keywords or phrases to target the desired audience, reaching the targeted customers, increase the number of visitors for your website for the targeted keyword(s) or phrase. Converting those visitors into potential customers is one of the arts of search engine optimization.

SEO is the only campaign which can derive targeted traffic through your website. Essentially more targeted traffic equal more sales. Once a website has been optimized, it will increase the visibility of your website in search engines. Within, brings a higher return on your investment than any other type of marketing for your company. This will therefore increase your volume of sales and profit overall.

Once a website obtains position through a SEO campaign, it should stay there for long term as opposed to PPC (Pay Per Click). SEO is a cheaper and long term solution than any other search engine marketing strategy.

8.9.7 Project Profitability

To calculate the profitability of this platform must be taken into account the income and annual expenses produced. Then is realized the study over a period of three years.

8.9.7.1 *Income*

The revenue model is based on revenues from media services group training and individual tutoring. Training for administrators in groups are 10 Euros and for individual counseling pay 30 Euros per week per customer. Marketing Service and First Contact are free.

Also there is a growth factor of 1.3% per year.

Table 7. Annual revenue model

Service							
	Traditional		**Price**	**Value**	**Unit**	**Quantity**	**Sum Euro in a year**
Media Service							
	Virtual training	Lecture Media	10	Euro	Weekly	120	62400
	Individual attention	Lecture Media	30	Euro	Weekly	90	140400
		Sum					202800
Marketing							
	First contact	Conferencing	0	Euro	Free	14	0
	Advertising	Advertising	100	Euro	Weekly	5	26000
		Bonus for Curs	-2	Euro	Weekly	44	-4611
		Sum					21389

Table 8.Revenue model for a term of 3 years.

Income				**2013**	**2014**	**2015**
Traditional				202800	202800	202800
First contact				0	0	0
Advertising				21389	0	0
Sum				224189	202800	202800

8.9.7.2 *Expenses*

For the calculation of costs should be taken into account both the construction of the platform as the platform running.

First starts of the required elements are:

- FTP server for data production and video server for drafting session.
- Production tools.
- System and drafting tools.
- File system.

- Web server (CMS, Streaming, Databases).
- Integration of the website.
- Network.
- Safety, quality, etc.

Once we have the necessary tools to develop of the multi-view platform on streaming, the costs will be reduced to the operating platform:

- Technical assistance.
- Hotline.
- Emergency and network failures.
- Server maintenance.
- Data security.
- Update and / or new versions of hardware and software.
- Testing and new customers.
- Technical quality control.
- Integration of new Internet services.
- Future benefits.

Table 9.Annual cost model.

Investments			Price	Value	Unit	Quantity	Sum Euro in a year
First Time							
	Network	Cables	10	Euro	Once	6	60
		Network Switch	300	Euro	Once	1	300
	Encoder	PC	1200	Euro		2	2400
		Software	600	Euro		2	1200
		Media: Cameras	12000	Euro	Once	1	12000
	Server	Webserver	3000	Euro		1	3000
		Streaming Server	3000	Euro		1	3000
		Sum					21960
Regular							
	Content	Lecture Media	3000	Euro	Monthly	12	36000
		Lecture Media	3000	Euro	Monthly	12	36000
		Lecture Media	4000	Euro	Monthly	12	48000
		Lecture Media	2500	Filme	Monthly	12	30000
		Lecture Media	2500	Filme	Monthly	12	30000
		Sum					180000
Handling							
	Network	DSL to Internet	800	Euro	Monthly	12	9600
	Operation	Employee Salary	25000	Euro	Once	1	25000
		Promotion	10000	Euro	Once	1	10000
		Sum					44600
Fixed							
	Service	Streaming	2000	Euro	Once	1	2000
		Lecture Media	0	Euro	Once	0	0
		Conferencing	0	Euro	Once	1	0
		Advertising	0	Euro	Once	25	0
		Documentation	200	Euro	Once	1	200
		Sum					2200

Table 10. Costs for a term of three years.

Cost	Year in Euro			2013	2014	2015
Investments				-21960	0	0
Regular				-18000	-18000	-180000
Handling				-44600	-44600	-44600
Fixed				-2200	-2200	-2200
Sum				-248760	-226800	-226800

8.9.7.3 *Product Benefit*

After the mathematical study done, we can see that the project is a complete business profitable. Here are the results obtained.

Table 11.Model Profitability.

Year in Euro	2013	2014	2015
Incom e	224189	291445	378880
Costs	-248760	226800	226800
Totals	-24571	64645	152080

8.9.8 Implementation Plan

So in this case the possible clients will be able to get started and familiar with the platform itself on- line, therefore a good example for each case that the platform can be produced, regarding the different market options should be presented as part of a web page, in order to present the product that will be offer to the company, though is not of interest to final users or customers. Nevertheless is also a way to show out to potential clients (companies) the interaction and recurrence of visitors that are attracted to the platform and show out the viable profits that the use of Multi Video can gain when launched.

The selling process is made by a B2B process since the creator of the platform will be the one offering to the Multi-View Video platform to companies willing to unleash a service or a product that will be finally delivered when the platform regarding their theme and needs in generated and then offered for the viewers to visit it. In this case the costumers that can be approached for commercial uses can be educative as for e-learning stages (language courses, on-line teaching, medical courses, company trainings), entertainment (sport games, concerts, theater plays) private or public security monitoring.

The product will arrive to the final costumers (B2C) when the company promotes the platform with their users and potential clients whether to use the platform as a service or to get the product promoted and market expansion for revenue forecast or sales revenue achieved.

8.9.9 Business Case for E- Learning Platforms Market

8.9.9.1 *Summary*

The E- Learning market is among the most rapidly growing sectors in the worldwide education and training industry. E-Learning holds an edge over other computer-based training initiatives due to its ability to lower costs by centralizing content, reduce logistics costs, facilitate persistent storage, and enable uniformity in content delivery.

An E-Learning platform for business can primarily be useful in training remotely home-based staff on new initiatives such as new campaigns for sales staff. It can also be an excellent method to deliver product training to new and existing customers, without the need for instructors to travel to your clients' location. In internal corporate training can offer an alternative stream of revenue by enabling long-distance learning as well as support for your existing face-to-face training courses.

Also is possible to take advantage of video streaming technology by saving lectures or tutorials particular to an area of learning. Many online courses use this strategy especially since teachers and students scarcely meet, if at all. When the participants record the lessons, they can review the content repeatedly at any convenient time. Thus using this technology provides a valuable advantage for enhancing distance education.

The global eLearning market is projected to reach $107.3 billion by the year 2015, driven by its benefits in the form of reduction in operational costs, flexibility in learning activities, and simplified training programs. The need to address the training and learning requirements of globally dispersed workforce is also driving the demand for eLearning programs.

8.9.9.2 *Analysis*

An E- Learning platform provides to people an innovative, attractive and fun way of learning tool directed to schools to implement their education system, by uploading in the different streams, the information they want to display, or by simply information storage that can be also modify and/or actualized in a very easy way. With this they can reinforce their education material and knowledge acquire in the classrooms and at the same time get to her students in a very practical, fun, usable and easy way. Also together with video streaming technology many online courses use this strategy especially since teachers and students scarcely meet, if at all. When the participants record the lessons, they can review the content repeatedly at any convenient time. Thus using this technology provides a valuable advantage for enhancing distance education.

8.9.9.3 *Strategies*

In order to have an adequate success with the E-Learning platform it's important to have a good direction in the material that is meant to be shown out, the objectives of the subjects regarding the program, the way in which is evaluated, tracked and reported.

The organization in the portal will be crucial as well, so the windows, supporting tools, and extra components should be settled since the beginning, as well as the links or references, they might have. When the whole idea is completed then the design of it should be establishes and consider the different ways to administrate the information so the security patterns can be established as well.

For the users willing to learn and gain more knowledge, to use an E-Learning platform is accessible, not need of extra cost, each one determines their pace, it goes according the own speed for learning, strengthens retention of information, they get consistent in the method and vision of the information.

8.9.9.4 *Goals*

The main objective and target of educative companies is to have the capacity of reaching more students/employees with virtual classes, allow instructors to teach from anywhere, record great classes for use indefinitely, track student participation and performance.

8.9.9.5 *Marketing Strategies*

An E- Learning platform is use the tool as support for their studies, reinforce and extend their knowledge and get familiar with it in daily use.

At the same time promotes the integration of Integrates with existing equipment, automates production and distribution process, is deployable across multiple campuses and finally goes on pursuit of offering the lowest cost per student per class solution; making of this the most attractive for people eager to learn.

8.9.9.6 *Implementation plan*

The Multi-View Video wants to be an interactive platform in the one the two popular tools for E-Learning are included, Blackboard Inc. and Moodle. Due to the support of a good quality Streaming Video, you can have access not only to the material, information that docents input, classes, homework support material in a very interactive, creative and friendly way. This will simplify the homework execution and track straight forward to self study at home and will be directed especially for students from 12, considering this the best range to interact with such interface.

By the Multi-View Video and the language selection option, have a better understanding and improve the understanding, saving cost s in the customers' different platforms creation and at the same time have a wider market acquire, since it can be from different speaking countries as well.

8.9.9.7 *Project Profitability*

An E- Learning platform is total feasible product, not only for the market vision and forecast that has in the market by itself, but at the same time for the opportunities that it offers as a programmer and platform designer, but as the market approach and therefore market customers gain that it generates.

Therefore is important to mention the Reduction in cost that it implies, since there is no need of a real studio or classroom in order for the class or assessment to take place. Due to this, administration and staff support and maintenance for the classes to take place should be dismissed as well from the investment and expenses budget.

Hence to the money that can be saved since the initial forecast of expenses, is possible to realize a larger expansion for gaining customers, with more catching attention and SEO investment while positioning and as well make it in a shorter time, making the expense start to reduce; making of this a total value for the educative companies.

9 Chapter 9

Production of a Single View Scenario

9.1 Introduction

A Single View Scenario, is a web page which hosts a single video. In this case, the video shows a class using an E- learning platform as a main tool.

In this chapter we will explain the process we follow to record the video and embed it in the web page. We will also explain why we choose e-learning platforms as the main topic of our video.

9.2 Why e-learning

Lately, education is changing and develops new methods of teaching, by using new resources such as the Internet. One of these developed methods is the use of e-learning platforms. E-learning is the computer and network-enabled transfer of skills and knowledge.

This method allows users to log on to the platform from any computer connected to internet. Only a username and a password are necessary.

E-learning platforms can be used as a supporting tool for a normal class, including interactive and/or multimedia content, texts, etc; or also as a main tool for distance learning.

Regarding to the use of streaming content, this platforms support video streaming, and as an example of its use, a teacher can record himself explaining the contents of a subject, and then upload it on a server, so the students can watch it via streaming from the platform.

Because e-learning platforms are one of the applications that use streaming content, we decided to show how a presence-class works when this method is used, not as a supporting tool but as a main tool for the development of the class.

9.3 Preparation for the recording

There are two parts in the filming process, a creative part and a technical part. In this phase we prepared both.

On the one hand, a storyboard had to be written and presented to Mr. Seidel, so he could correct it. This storyboards content is a small review of how the film is going to be.

A short visit to the school was required, in order to get an idea of the space and the material that would be required for the recording.

Figure 20.First visit of High School Hans- Carossa

Because the whole recording had to be done in just one day, we made a schedule and adjusted it to the timetable of the students and teachers.

Table 12. Schedule of the recording

Recording the Lo-net-Film

Day of recording: Thursday 29.03.2012
Carossa OS, Berlin Spandau

Schedule:

9:00 Beuth-HS (Picking up the equipment)
10:00 Carossa-OS
10:20 – 11:20 (–Schule, Pause etc.)
12:00 - 12:45 Prepairing Room (lighs etc.)
12:45 – 14:20 Lesson (working with 2 cameras)
14:20-14:45 Disassemble the equipment
14:45 – 17:00 Recording with Hr. Rüßbült in the library.

Additional information

*The additional information is shown as it was sent.

Table 13. Aditional Information

Content agreements

Ziel:

Lehrer, die lo-net nochnieeingesetzthabensolleneinenEindruck von Planung, Durchführung und Nachbereitung des Unterrichtserhalten.

WelcheKompetenzenwerdenmitdieser Art Unterrichtabgedeckt, was ist das eigentlichNeue?

- Selbstverantwortlich
- arbeiten in Teams
- Nutzungneuer Medien mitentsprechendenKompetenzen (Recherche, TexteÜberfliegen, Inhalte/Ergebnisse digital zusammenstellen)
- KomplexereAufgabenstellungenbearbeiten (Problemlösekompetenzen)

Die Nutzung der PlattformbringtfolgendeVorteile:

- AdäquateLerntypenansprache
- VerschiedeneMaterialien
- Lehrer kannErgebnissesoforteinsehen und ggf. direktnoch in der Stunde Feedback an die Erstellergeben („esfehltnoch dies oder das“). Ansonstenkanner die ErgebnissekommodezuHauseaufrufen und istnicht an die Situation vor Ort gebunden
- Schülerergebnissekönnenan die anderenSchülerzurBewertungweitergereichtwerden.
- EswerdenbegleitendweitereInformationenoderVertiefungenangeboten (zusätzlicheMaterialien). JederSchülernimmtsich die Hilfe/Dokumente, die erbraucht.

JederSchülerfindetfürsicheineVereinfachung des komplexen Systems.DieseVereinfachungist das Lernergebnis.

Vorbereitung

Der Lehrer recherchiertzudemThema

FindetgeeigneteAdressen, die eralsVertiefungmitzu den Inhaltendazugibt.

Durchführung

KlareAufgabenstellung und Zeitvorgabe.KomplexeAufgabenstellung

4 Grundthemen

Organisation:

Partnerarbeit an einem der 4 Themen pro Team

Ergebnissicherung:

NeueZusammensetzung: 4er Teams in dem je einereinThemabearbeitet hat. Jeweilsstelltjeder die Lösung seiner Aufgabe den anderendar.

ZusätzlichetechnischeHerausforderung:

ImRahmen der Bachelorarbeitsollen

1. Einkurzer Film für den Verlagentwickeltwerden, der dann auf der lo-net-Webseiteveröffentlichtwird.
2. EinFilmwerk, welches Multiview (parallel abspielendeunterschiedlicheEinstellungen) enthält. HierzusindparalleleAufnahmennotwendig. Totale, Halbtotale, Emotion, Screen (angeschnitten).

Inhaltlich können das sein:

- Totale und Monitor
- Schüleraktion und Kommentar/didaktische Implikationvom Lehrer
- Unterschiedliche Herangehensweisen der Schülergruppenaneine Problematik

Das Multiview-Werkistnur auf den Webseiten der Beuth-Hochschulezusehen. Von den Seiten von lo-net wird es einenPartnerlinkauf das Multiview-Werkgeben.

9.3.1 Material Used

The following material was used during the recording:

Table 14. Material used for the film

Material	Model	Main Characteristics	Photo
Video camera	Sony HDR-FX1000E	HDV/DV 1080i	
Microphone	T. Bone TWS Lavalierset 863MHz	Wireless lapel microphone	
Camera	Canon EOS 450D	8-Megapixels Objective: 17-85mm	
2 Light set		2x800Watt Video lights	
Video camera tripod	Manfrotto 128RC		

9.3.2 Recording the film

As mentioned before, the video recording was done at the High School Hans-Carossa in Spandau.

First, a class using the e-learning platform Lo-net2 was filmed and after that a short interview as recorded in the library between Mr. Rußbült (the expert teacher from the platform) and Mr. Thomas Seidel (one of the persons who are in charge of the platform).

The film was recorded as planned. The 2 hours lesson was recorded without interruptions, to make sure that the environment of this class was as authentic as possible. Therefore, tripods were not used; we were constantly moving to keep the picture free of technical support such as lights, cameras and microphones.

During the interview, one of the cameras was static, while the other one was constantly moving around to see the interview from different perspectives.

Because of the above mentioned points, this part was easier to record and not many takes were necessary.

In order to develop the streaming single view into a streaming multi-view, two video cameras were used during the recording.

Figure 21. Preparing the material

Figure 22. Testing the material

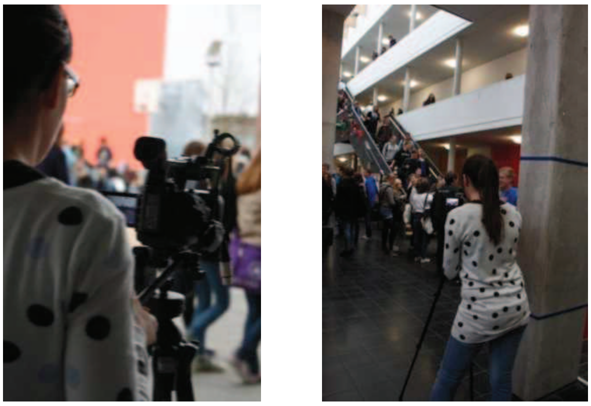


Figure 23.General view of the High school

Figure 24. Recording the class

Figure 25.Recording the interview

9.4 Post-Production

9.4.1 Introduction

The post- production phase of creating a film usually takes longer than the actual shooting of the film, because it includes the complete editing, color correction and the addition of music and sound. Opening and/or end credits, optical and visual effects must also be added.

9.4.1.1 *Edit*

The process of editing a movie is also seen as the second directing, because through the post production it is possible to change the intention of the movie. Editing turns individual scenes, called raw footage, into a finished motion picture. Editors splice all of the usable footage together into a coherent storyline according to the storyboard.

The process is basically always the same; first the images are loaded onto a computer using a program (in our case Adobe Premiere Cs5). With this program, it is easy to try different ways of arranging the shots, mainly in the order specified by the shooting script.

In our film, we intercalated scenes from the class in between scenes from the interview, which describes the events happening during the class. To provide dynamism to the film, multi-camera tool was used. It allows us to combining 4 scenes at the same time by recording the selected scenes and producing a new scene with the chosen pieces.

Figure 26.Editing the film using multi-camera tool

In order to explain to the viewers what they could expect from the development of the multi-view scenario, we divided the screen in our project into 4 parts and showed the scenes in the single view scenario.

Figure 27. Editing the film. View as a Multi-view scenario

Header and Introduction were made with Adobe After Effects Cs5. The designs of them follow the corporate design of Lo-net2.

The audio also had to be edited. Increasing volume and noise reduction, add parametric equalizers and changing the Threshold of Dynamics were used to increase the sound quality.

Finally audio and video effects were added, just to smoothen the changes between scenes and questions.

9.5 Export /Convert

In the final step of the postproduction, we had to export the edited scenes into one single file, the final film.

We have to take into consideration to who the video is destined to, so it can be converted into the appropriate format.

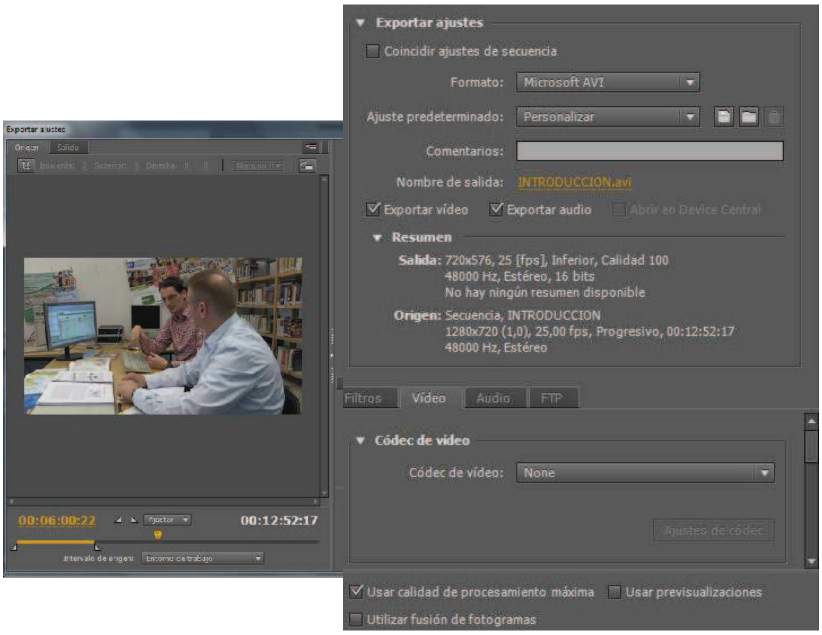


Figure 28.Exportingthe file. Settings

Problems with the post production.

One of the main problems on this stage was exporting the video with the highest quality as possible.

On a first try, the video was codified with an H-264 format, in which with the naked eye there are no visual disturbances. In the next stage, which is the encoding part, we cannot use the H-264 format anymore. The problem is that this format already compresses the video, using 4:2:2 chroma sampling, so when we encoded the video for the streaming platform, the result of the video was the following

Figure 29. Problems with the post-production. Bad quality of the film

In order to have a non compressed file, the next choice was Microsoft AVI non-compressed, but it also does not work properly. Using V22C, the file cannot be played in normal players like VLC or Windows Media Player; using YUVUC, only the video was converted, but not the audio. The next solution was using the format Microsoft AVI, without compressor, this option was a good solution, the quality of the AVI video and the Encoded video is acceptable, not pixilated images, but the problem is the size of the frames, it does not fit properly on the browsers player. This problem is caused because with this format, the size of the frames is set and cannot be changed.

The last and final solution was the option “Match sequence settings” which maintain the properties of the original videos, the final format is MPEG. Using this format the next steps presented no problems of quality or reproduction. The quality is as good as the archived with the AVI format.

9.5.1.1 *Encoding*

As it was mentioned before, the next step was encoding the files. That means, compressing the film in different formats, so it can be played in all browsers.

The program used was FFmpeg, an open source program, which includes libavcodec - the leading audio/video codec library.

It is important to have the file compressed in different formats to avoid problems due to compatibly issues with different browsers.

Therefore the formats used are:

- Theoraogg, specified for HTML 5 by W3C, which works with Firefox and Opera.
- Format webm(vp8) for Chrome mainly
- And format H264 (mp4) for Chrome, Explorer and Safari.

The 3 files used by ffmpeg to convert the files are:

_encode_ogg: Bloc de notas

Archivo Edición Formato Ver Ayuda

```
rem ffmpeg script: create ogg
rem uses ffmpeg presets
rem 2012,

rem input/output files
set input="in/D.mpeg"
set output="out/G.ogg"
rem length in seconds (optional)
 set timelength=-t 25


bin\ffmpeg.exe  -threads 4 -i %input% -b 2000k -s 720x404 %output%

goto end

:err
echo Presets not installed - please check ffmpeg_presets.bat

:end
pause
```

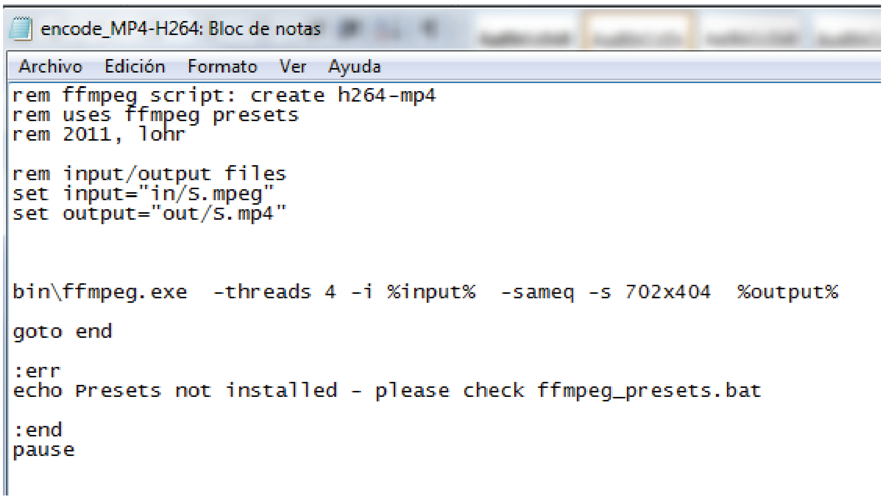
Figure 30.Code for ogg format

_encode_webm: Bloc de notas

Archivo Edición Formato Ver Ayuda

```
rem ffmpeg script: create webm
rem uses ffmpeg presets
rem 2011,

rem input/output files
set input="in/D.mpeg"
set output="out/D.webm"

bin\ffmpeg.exe  -threads 4 -i %input%  -s 720x404 -b 2000k -vcodec libvpx -r 25 %output%

goto end

:err
echo Presets not installed - please check ffmpeg_presets.bat

:end
pause
```

Figure 31.Code for WebM format


encode_MP4-H264: Bloc de notas

Archivo Edición Formato Ver Ayuda

```
rem ffmpeg script: create h264-mp4
rem uses ffmpeg presets
rem 2011, lohr

rem input/output files
set input="in/S.mpeg"
set output="out/S.mp4"


bin\ffmpeg.exe  -threads 4 -i %input%  -sameq -s 702x404  %output%

goto end

:err
echo Presets not installed - please check ffmpeg_presets.bat

:end
pause
```

Figure 32.Code for H264 format

Problems during the encoding process:

As said before, the main problem in this stage is that the codified files could not have the necessary quality to be streamed; this can occur for 2 reasons. It can be because the source format is already compressed, and compressing it again will always affect the quality. To solve it, we had to step back to the converting phase and try another format without compression.

The other reason can be because of the bit rate. On a first try we used the option –sameq, so the output file preserves the same quality as the input file, but this option only worked out for H264 (mp4).

For the other format types, we modified the bit rate. For webm we increased the bit rate as much as possible (500Kb), for ogg, 2000Kb was enough.

9.6 Loading the video on the web

The last step is creating the web page; load the page and the files into a fileserver so they are visible to anyone.

Most videos are shown through a plug-in, but thanks to HTML 5, which defines a new element (<video>), that specifies a standard way to embed a video on a web page, we can load videos in an easy way.

Internet Explorer 9, Firefox, Opera, Chrome, and Safari support the <video> element.

9.6.1 Code

9.6.1.1 *HTML*

The source code of the web page is located at Annex 1.

9.7 Results

Following the final scenario is shown.

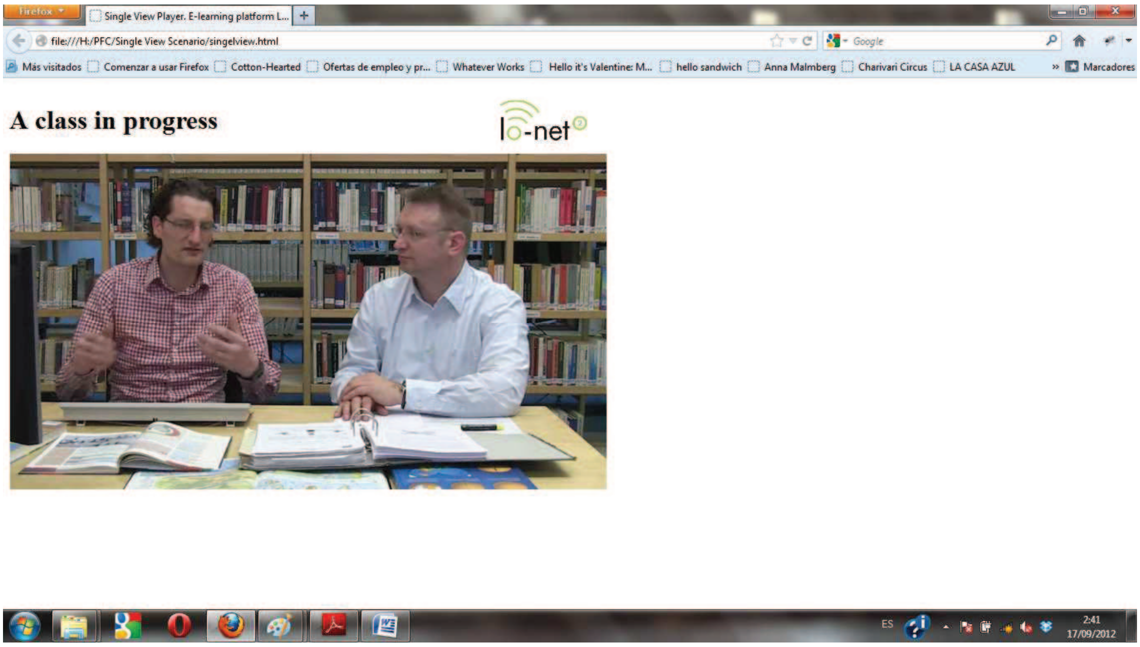


Figure 33. Final result of a Single-view scenario

10 Chapter 10

Production of a Multi-view Scenario.

Changing from Single view to Multi-view

10.1 Introduction

The changing from Single view to Multi-view scenario means using the same resources used from the Single view to build the Multi view scenario. Except from recording, the next steps are different from the ones taken for the Single-view, also its purposes and finalities searched.

10.2 Recording the film

As mentioned before, in order to work also with a Multi view scenario, two cameras were used. Some scenes were recorded by both cameras at the same time, to show the synchronization between videos in the scenario.

No special measures were taken to record the audio, because all the possible changes will be done in the post production process.

10.3 Post-Production

The post- production of a Multi-view scenario has to be carefully done. One of the most important aspects is the synchronization between videos.

This time, 4 videos were edited, and because there were two parts in the recording, two of the videos will refer to the lesson and the other two will refer to the interview.

We had chosen a few questions from the interview, and then selected the scenes of the class which are related to these questions and basically give an answer.

We synchronized the scenes with Adobe Premiere tools.

Figure 34. Synchronizing videos

Regarding the audio, we have three different languages to choose: German, English and Spanish. The original language of the video is German; the other two languages had to be folded by two people each, the interviewer and the interviewee.

For this, was used a cardioid condenser microphone with USB digital output. After the recording, the audio was edited using parametric equalizers (D. Noiser and amplification) and dynamic processors such as compressors (Threshold). Thereby background noise was eliminated, the hiss in the voice, volume, etc. This process is also done with Adobe Premiere tool.

The default audio is German, which is included in the first video. The English audio is included in the second video and the Spanish audio is included in the third one. The fourth and fifth videos have not audio. This is the control through which we can choose the language that we want to listen.

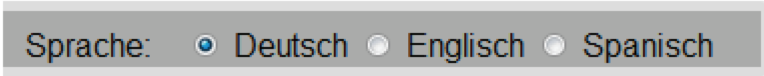


Figure 35.Audio selection control

After this, the process is the same like the process of a single view scenario.

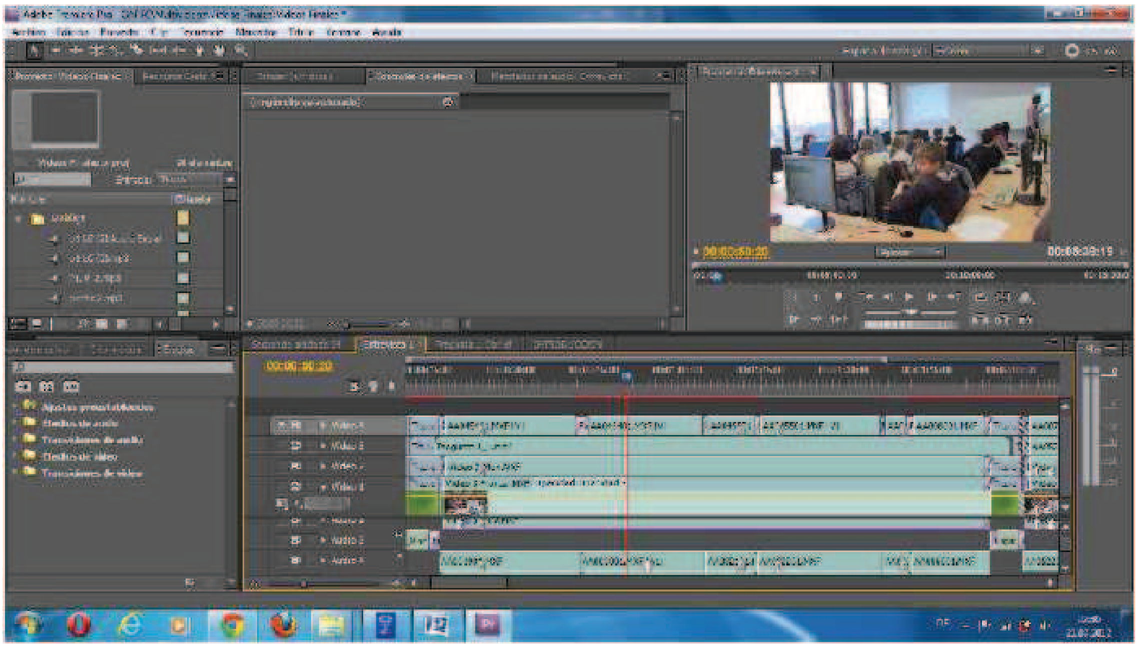


Figure 36.Edited video

10.4 Code HTML5

The source code of the web page is located at Annex 2.

10.4.1 Choice of audio

When playing the videos, as mentioned before, the default audio is German. Like all are played at the same time, what happens is that is listened the first video, and the rest are positioned on mute. When the English option is chosen, the first video and third video are positioned on mute, and the second one would begin to listening. In the same way happens with the Spanish language, in which would hear the third video, and the first two would be on mute.

In the following, the part of the code that performs the stated above is shown:

Listing 7. Audio script

```
        <input type="radio" name="opciones" value="Deutsch"
checked="checked"   onclick="video2.muted=true; video3.muted=true;
video.muted=false;"/> Deutsch

        <input type="radio" name="opciones" value="Englisch"
        onclick="video.muted=true; video2.muted=false;
video3.muted=true;"/> Englisch

        <input type="radio" name="opciones" value="Spanisch"
        onclick="video.muted=true; video2.muted=true;
video3.muted=false;" /> Spanisch
```

For this we have used the *input* tag that is used primarily to create forms. We created the *radiobutton* type controls, which are used when the user can select a single choice between related options that are presented. The value of the *type* attribute for these form controls is *radio*. The *name* attribute is used to indicate to the *radiobutton* that they are related. Therefore, when several *radiobutton* have the same value in its *name* attribute, the browser knows that they are related and it can deselect an option when other one is selected.

To associate the *onclick* event to the input tag of XHTML, is used JavaScript. The code is included as an attribute of the own XHTML element. The event consists in clicking and releasing the mouse, and then, the code included in the event starts running. So when you click on *value* = English, the second video will begin to hear while that the first one and third videos will positioned on mute.

10.5 Problems encountered and solutions taken

10.5.1 Type MIMES

There is still debate amongst browser vendors and web developers as to which, if any, video format should be the standard format for HTML5 video. Currently, both Firefox and Chrome support the Ogg Theora format, while Chrome supports H.264 MP4 and Firefox does not. Meanwhile, Microsoft has indicated a preference for supporting the H.264 MP4 format in IE 9. Because of these compatibility problems with the different browsers and video formats, is used the MIME type.

MIME is a standard that classifies resources and provides information (to the programs) about how to handle them. This allows the proper handling and interpretation of different types of files by programs (such as browsers). For example, thanks to MIME, browsers can successfully open a file ".html" as a source of web page and not like a plain text, a video or other.

When a MIME type is not specified for a resource, the program that handles it, can "guess it” from the extension (for example, a file with extension ".bmp" should contain a bitmap image). But this may not always be successful as a single extension can be associated with more than one format. Meanwhile, MIME types are unique. This is the main reason for using MIME types whenever possible.

In HTML documents, authors can use MIME types in many instances, usually through the "type" attribute. Some special cases of its use are the attributes "enctype" of the HTML form element, and with attribute "content-type" of the HTML meta tag.

In the element "source" in addition to the address of the file, must be specified the MIME type of format with the attribute type = "video / format" for the videos and type = "audio / format" for audio. As mentioned there are three standards and each supports a different format H.264 (mp4, mp3 and acc), Theora / Vorbis (ogg and ogv) and WEBM (WebM).

Types in video:

```
type="video/ogg"
type="video/webm"
type="video/mp4"
```

Types in audio:

```
type="audio/ogg"
type="audio/mpeg"
type="audio/webm"
```

Here is shown, how we have used the MIME type for the video two:

Listing 8. MIME type for the video two.

```
<video tabindex="0" autobuffer id="two">
    <source src="assets/E.webm" type="video/webm" >
    <source src="assets/E.ogg" type="video/ogg" >
    <source src="assets/E.mp4" type="video/mp4" >
</video>
```

10.5.2 Fallback options

Older browsers are unable to support or play a video tag. But this problem is fixed by including fallback content in the video tag. A Flash code is included into the fallback content which enables old browsers to play the video with Flash when a browser is not compatible with HTML5. A transcript of the video is included into the fallback content in order to allow incompatible browsers users to read a transcript of the video.

Fortunately, we have not had to use the fallback options because our code works perfectly in Firefox. Anyway, we have found it interesting and helpful for people who start programming with HTML5. Below an example is showed:

Listing 9. Fallback options example.

```
<video src="video.ogv" controls>
    <object data="flvplayer.swf" type="application/x-shockwave-flash">
      <param value="flvplayer.swf" name="movie"/>
    </object>
</video>
```

10.6 Final Results

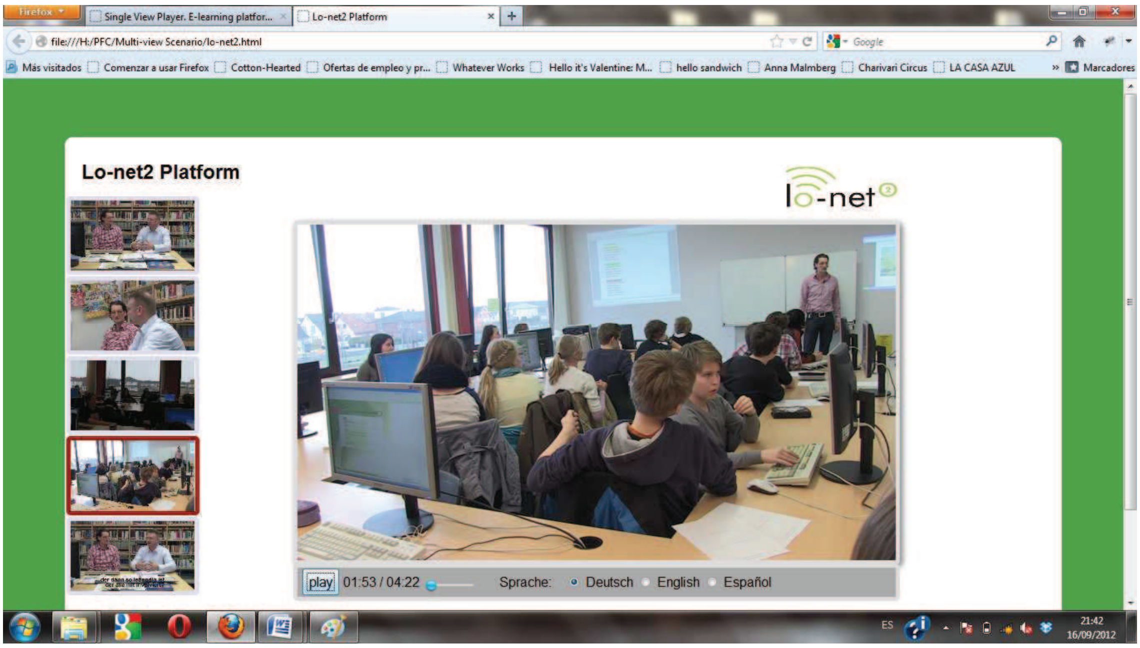


Figure 37. Multi-view video platform

Figure 38. Multi-view video platform

11 Conclusions

The previous research accomplished on this thesis, have made possible the understanding of the new technologies of video delivering through the Internet, mainly focused on the streaming process, Multi-view systems and the video post-production of both Single-view scenario and a Multi-view Scenario.

The need concepts in terms of HTML5 and JavaScript for web programming have been obtained in order to improve the Multi-view video scenario, achieving one of the main targets, which is to embed the audio option.

As well, the functioning of programs such as Video Adobe Edition Premiere CS5 and FFmpeg for codifying has been learned. It is important to remark that AP CS5 is one of the most outstanding editing softwares in the market.

The main objective in the first chapter of this thesis has been achieved satisfactorily. Since the Single-view scenario as the Multi-view scenario functions perfectly, their videos are synchronized joint with the audio option and subtitles. Furthermore, the Multi-view platform can be commercialized as it has been show out in the Business Case.

Finally it can be established that for future studies new application can be developed and generated such as the embedding subtitles out of the display, allowing the addition of another video signal, even considering the multiple selection of subtitles and audio at the same time. Another suitable option can be using the Chroma Key application.

13 Bibliography

- Apostolopoulos John G., Wai-tian Tan and Wee, Susie J., *Video Streaming: Concepts, Algorithms and System*s, Mobile and Media Systems Laboratory, HP Laboratories, Palo Alto HPL-2002-260, September 18th 2002 [online]

- Austerberry David, "The Technology of Video & Audio Streaming", Second Edition. UK: Elsevier, 2005.

- Bedall*, P.*, Streaming Media: The New Essential 2012. [online] <http://ezinearticles.com/?Streaming-Media:-The-New-Essential&id=7000193>

- Chunlei,Liu., "Multimedia Over Ip: RSVP,RTP, RTCP, RTSP" 02.07.2000 <http://www.cis.ohio-state.edu /˜jan/cis788-97/ip_multimedia/indext.html

- Dapeng, Oliver Wu., "Multi-view Image Processing:Correct State of the Art, Challenges and Future Directions" Electrical and Computer Engineering. University of Florida, USA. 18 August 2006. [online] http://www.wu.ece.ufl.edu/

- Evans, C. & Gibbons N.J., The interactivity effect in multimedia learning. Volume 49(4), Computers & Education, December 2007, pp. 1147–1160.

- García Peñalvo, F.J., Estado actual de los sistemas e-learning [online]. Teoría de la educación: educación y cultura en la sociedad de la información. Universidad de Salamanca, volumen 2005 - 6 (2). [online]

  <http://campus.usal.es/~teoriaeducacion/rev_numero_06_2/n6_02_art_garcia_penalvo.htm>

- Hallivuori, Ville, "Real-Time Transport Protocol (RTP) security". Seminar on Network Security. Helsinki University of Technology, Finland.2000

- Koistinen, Tommi, "Protocol overview: RTP and RTCP" Nokia Telecommunications. [online]

- Kabota, Akira et al. "Multi-view Imaging and 3DTV". 27 July 2007. [online]

- Lohr, Jürgen, "High Definition Media Services."Germany. Schiele und Schön.2009

- Ozer, Jan, "What is Streaming?". A high-level view of streaming media technology, history, and the online video market landscape. February 26, 2011. [online]

<http://www.streamingmedia.com/Articles/ReadArticle.aspx?ArticleID=74052&fb_source=message>

- Perkings, C., ”An Introduction to RTP, in its: RTP: Audio and Video for the Internet.” Addison Wesley, 12 June 2003.
- Perkings, C., ”Voice and video Communication over Packet Networks, in its RTP: Audio and Video for the Internet.” Lugar, Addison Wesley, June 12, 2003.
- Pfeiffer, S., ”The Definitive Guide to HTML5 video.” United States of America, Apress, December 29, 2010.
- Rijo Sciara, Daniel. “Fundamentos de Video Streaming”. Facultad de Ingeniería. Universidad de la Republica de Montevideo, Uruguay. Dec 2004
- Simpson, W., “Video over IP.” Second Edition, UK, Focal Press, August 14, 2008.
- Yo-Sung, HO & Kwan-Jung Oh, “Overview of Multi-view video Coding”.Gwangju Institute of Science and Technology (GIST),Korea. 2007.[online]

14 Annexes

14.1 Annex 1.

14.1.1 HTML5 Code for the Single-view Scenario.

```
<!DOCTYPE HTML>

<html>

<head>

      <title>Single View Player. E-learning platform Lo-net2</title>

      <meta http-equiv="Content-Type" content="text/html; charset=utf
      8" />

      <link rel="stylesheet" href="Dateien/singleviewstyle.css"
      type="text/css">

</head>

<body>

      <div class="header">

           <a id= "web" href= "http://www.lo-net2.de"> <img id="logo"
           src="Dateien/lonet2.jpg"></a>

            <h1>A class in progress</h1>

      </div>

      <video  width="720" height="404" controls>

            <source src="assets/single view.webm" type="video/webm" />

            <source src="assets/single view_ogg.ogg" type="video/ogg"/>

            <source src="assets/single view_H264.mp4"type="video/mp4"/>

            This browser is not compatible with HTML 5

      </video>

</body>

</html>
```

14.1.2 CSS code for the Single-view Scenario.

```
.header{

 width: 700px;

 height: 70px;

}

h1 {

font: "Helvetica Neue", Helvetica, sans-serif;

  padding-top: 10px;

}


 #web {

  float:right;

}


#logo{

 height: 60px;

float:left;

 }
```

14.2 Annex 2.

14.2.1 HTML5 Code for the Multi-view Scenario.

```
<!DOCTYPE HTML>

<html lang="en"><head>

<meta http-equiv="content-type" content="text/html; charset=UTF-8">

<meta name="viewport" content="width=620">

<title>Lo-net2 Platform</title>

<link rel="stylesheet" href="two-videos-Dateien/html5demos.css"

type="text/css">

<link rel="stylesheet" href="Dateien/multiviewstyle.css"
type="text/css">

<link rel="stylesheet" type="text/css" media="screen, projection"
href="./css/fd-slider.css">

<script src="two-videos-Dateien/h5utils.js"></script></head>

      <!-- Load JQuery and JQuery-UI -->

 <script type="text/javascript" src="js/jquery 1.4.2.min.js"></script>

      <!-- Load Processing.js -->

 <script language="javascript" src="js/processing.js"></script>

 <script language="javascript" src="js/init.js"></script>

 <script src="js/fd-slider.js"></script>

 <script>

      fdSlider.createSlider({

      inp:document.getElementById("scrub"),

      step:1,

      maxStep:10,

      min:1,

      max:262,

      animation:"tween",

      forceValue:false});
```

```
</script>

<body style="background:#51A34B";class="">

      <div id="fondo"></div>

      <div class="header">

            <a id= "web" href= "http://www.lo-net2.de"> <img id="logo"
      src="Dateien/lonet2.jpg"></a>

      </div>

      <section id="wrapper">

          <header>

            <h1>Lo-net2 Platform</h1>

          </header>

      <style>

            video {

                  display: block; /* styling shim for older browsers */

                  top: 2px;

                  bottom: 20px;

                  left: 330px;

                  width: 150px;

                  padding: 0;

                  margin: 2px;

                  border: 5px #dde solid;

                  border-radius: 5px;

            }

            #controls {

                  position: absolute;

                  top: 563px;

                  bottom: 20px;

                  left: 330px;

                  border-radius: 5px;
```

```
        border: 5px #ddd solid;
        background: #aaa;
        width: 720px;
        height: 35px;
}
#canavas_i{
        position: absolute;
        top: 170px;
        left: 350px;
        bottom: 20px;
        width: 720px;
        height: 404px;
        padding: 0;
        margin: 0;
        border: 5px #ddd solid;
        border-radius: 5px;
        box-shadow: 3px 3px 7px #777;
}
#vs1, #vs2, #vs3, #vs4, #vs5 {
        position: absolute;
        bottom: 60px;
        left: 8px;
        width: 148px;
        height: 80px;
        padding: 0;
        margin: 0;
        border: none;
        background: transparent;
}
```

```
        #vs1 {
              top: 120px;
        }
        #vs2 {
              top: 230px;
        }
        #vs3 {
              top: 310px;
        }
        #vs4 {
              top: 425px;
        }
        #vs5 {
              top: 505px;
        }
  </style>
  <article>
         <p id="canavas_i">
              <canvas id="c1_id" width="720" height="404">
        </canvas>
          </p>
  </article>
  <article >
         <video tabindex="0" autobuffer id="one">
              <source src="assets/E.webm" type="video/webm" >
              <source src="assets/E.ogg" type="video/ogg" >
              <source src="assets/E.mp4" type="video/mp4" >
        </video>
```

```
        <video tabindex="0" autobuffer id="two">
            <source src="assets/E.webm" type="video/webm" >
            <source src="assets/E.ogg" type="video/ogg" >
            <source src="assets/E.mp4" type="video/mp4" >
        </video>

        <video tabindex="0" autobuffer id="three">
            <source src="assets/S.webm" type="video/webm" >
            <source src="assets/S.ogg" type="video/ogg" >
            <source src="assets/S.mp4" type="video/mp4" >
        </video>

        <video tabindex="0" autobuffer id="four">
            <source src="assets/V4.webm" type="video/webm" >
            <source src="assets/V4.ogg" type="video/ogg" >
            <source src="assets/V4.mp4" type="video/mp4" >
        </video>

        <video tabindex="0" autobuffer id="five">
            <source src="assets/A.webm" type="video/webm" >
            <source src="assets/A.ogg" type="video/ogg" >
            <source src="assets/A.mp4" type="video/mp4" >
        </video>
</article>
<article>
       <p id="v_selector">
            <button  id="vs1" onclick="vs1()"> </button>
            <button  id="vs2" onclick="vs2()"> </button>
            <button  id="vs3" onclick="vs3()"> </button>
```

```
                <button  id="vs4" onclick="vs4()"> </button>

                <button  id="vs5" onclick="vs5()"> </button>

            </p>

    </article>

    <article>

           <p id="controls">

                <input id="play" value="play" type="button">

                <span id="position">00:00</span> / <span
                id="duration">00:00</span>

                 <input id="scrub" max="256" value="1" type="range">

                    Sprache:   

                <input type="radio" name="opciones" value="Deutsch"
                checked="checked" onclick="video2.muted=true;
                video3.muted=true; video.muted=false; "/> Deutsch

                <input type="radio" name="opciones" value="English"
                onclick="video.muted=true; video2.muted=false;
                video3.muted=true;"/> English

                <input type="radio" name="opciones" value="Espaniol"
                onclick="video.muted=true; video2.muted=true;
                video3.muted=false;" /> Español

            </p>

    </article>

    <div id="Status">

          <div id="display"></div>

            </div>

 <!--

    Sliders created using the Javascript API from here on in i.e. no
    more HTML5 range polyfills from this point on.

-->



      <div>
```

```
        <!-- Here's the form element we will associate the third
slider with.

        <input  id="range1"     /> -->

    </div>

<script>

          var video = document.querySelector('#one'),

          video2 = document.querySelector('#two'),

          video3 = document.querySelector('#three'),

          video4 = document.querySelector('#four'),

          video5 = document.querySelector('#five'),

          togglePlay = document.querySelector('#play'),

          position = document.querySelector('#position'),

          ready = false,

          controls = document.querySelector('#controls'),

          scrub = document.querySelector('#scrub');

          c1 = document.getElementById('c1_id');

          c1tx = c1.getContext('2d');

          video_main=video;

          video.style.border = "5px #900 solid" ;

          setInterval("processFrame()", 33);

          video2.muted = true;

          video3.muted = true;

          video4.muted = true;

          video5.muted = true;




      addEvent(togglePlay, 'click', function () {
```

```
  if (ready) {
      video.playbackRate = 0.5;
   if (video.paused) {
   if (video.ended) {
        video.currentTime = 0;
        video2.currentTime = 0;
        video3.currentTime = 0;
        video4.currentTime = 0;
        video5.currentTime = 0;
      }
      video5.currentTime = video.currentTime;
      video4.currentTime = video.currentTime;
      video3.currentTime = video.currentTime;
      video2.currentTime = video.currentTime;
      video.play();
      this.value = "pause";
    } else {
      video.pause();
      this.value = "play";
    }
  }
});

function seek() {
  scrub.value = video2.currentTime = this.currentTime;
  scrub.value = video3.currentTime = this.currentTime;
  scrub.value = video4.currentTime = this.currentTime;
  scrub.value = video5.currentTime = this.currentTime;
}
```

```
addEvent(video, 'seeking', seek);
addEvent(video, 'seeked', seek);
addEvent(video, 'play', function () {
  video2.play();
  video3.play();
  video4.play();
  video5.play();
});

addEvent(video, 'pause', function () {
  video2.pause();
  video3.pause();
  video4.pause();
  video5.pause();
})

addEvent(video, 'timeupdate', function () {
  position.innerHTML = asTime(this.currentTime);
  scrub.value = this.currentTime;
});

addEvent(video, 'ended', function () {
  togglePlay.value = "play";
});

addEvent(video, 'canplay', function () {
//  video.muted = true;
  ready = true;
```

```
  document.querySelector('#duration').innerHTML =
asTime(this.duration);


  scrub.setAttribute('max', this.duration);

  addEvent(scrub, 'change', function () {

    video.currentTime = this.value;

    video2.currentTime = this.value;

    video3.currentTime = this.value;

    video4.currentTime = this.value;

    video5.currentTime = this.value;

  });

});


function asTime(t) {

  t = Math.round(t);

  var s = t % 60;

  var m = Math.round(t / 60);


  return two(m) + ':' + two(s);

}


function two(s) {

  s += "";

  if (s.length < 2) s = "0" + s;

  return s;

}


function vs1() {

 video_main.style.border = "5px #dde solid" ;
```

```
 video_main=video;
 video.style.border = "5px #900 solid" ;
}

function vs2() {
 video_main.style.border = "5px #dde solid" ;
 video_main=video2;
 video2.style.border = "5px #900 solid" ;
}

function vs3() {
 video_main.style.border = "5px #dde solid" ;
 video_main=video3;
 video3.style.border = "5px #900 solid" ;
}

function vs4() {
 video_main.style.border = "5px #dde solid" ;
 video_main=video4;
 video4.style.border = "5px #900 solid" ;
}
function vs5() {
 video_main.style.border = "5px #dde solid" ;
 video_main=video5;
 video5.style.border = "5px #900 solid" ;
}


function processFrame(){
```

```
                        c1tx.drawImage(video_main, 00, 00,720,404);

            }
      </script>
      </section>
</body>
</html>
```

14.2.2 CSS Codes for the Multi-view Scenario.

14.2.2.1 *Html5demos.css*

```
body {
  font: normal 16px/20px "Helvetica Neue", Helvetica, sans-serif;
  background: rgb(237, 237, 236);
  margin: 0;
  margin-top: 0px;
  padding: 0;
}

section, header, footer {
  display: block;
}

#wrapper {
  width: 1200px;
  margin: 0 auto;
  background: #fff url(/images/shade.jpg) repeat-x center bottom;
  -moz-border-radius: 10px;
  -webkit-border-radius: 10px;
  border-radius: 10px;
```

```
  border-top: 1px solid #fff;
  padding-bottom: 76px;
}

h1 {
  padding-top: 10px;
}

h2 {
  font-size: 100%;
  font-style: italic;
}

header,
article > *,
footer > * {
  margin: 20px;
}

footer > * {
  margin: 0px;
  color: #999;
}

#status {
  padding: 5px;
  color: #fff;
  background: #ccc;
}
```

```
#status.fail {
  background: #c00;
}

#status.success {
  background: #0c0;
}

#status.offline {
  background: #c00;
}

#status.online {
  background: #0c0;
}

footer #built:hover:after {
  content: '...quickly';
}

[contenteditable]:hover {
  outline: 1px dotted #ccc;
}

abbr {
  border-bottom: 0;
}
```

```
abbr[title] {
  border-bottom: 1px dotted #ccc;
}

li {
  margin-bottom: 10px;
}

#ffad {
  font-size: 90%;
  border: 1px solid #ccc;
  background: #fcfcfc;
  display: block;
  -moz-border-radius-topleft: 25px;
  -webkit-border-top-left-radius: 25px;
  -moz-border-radius-bottomright: 25px;
  -webkit-border-bottom-right-radius: 25px;
  border-top-left-radius: 25px;
  border-bottom-right-radius: 25px;
  color: #000;
  text-decoration: none;
}

#ffad:hover {
  border-color: #919191;
}

#ffad section {
  padding: 20px;
```

```
}

#ffad p {
  margin: 10px 10px 10px 100px;
}

#ffad img {
  border: 0;
  float: left;
  display: block;
  margin: 14px 14px 0;
}

#ffad .definition {
  font-style: italic;
  font-family: Georgia,Palatino,Palatino Linotype,Times,Times New
Roman,serif;
}

#ffad .url {
  text-decoration: underline;
}

input {
  font-size: 16px;
  padding: 3px;
  margin-left: 5px;
}
```

```
#view-source {
  display: none;
}

body.view-source {
  margin: 0;
  background: #fff;
  padding: 20px;
}

body.view-source > * {
  display: none;
}

body.view-source #view-source {
  display: block !important;
  margin: 0;
}

#demos {
  width: 560px;
  border-collapse: collapse;
}

#demos .demo {
  padding: 5px;
}

#demos a {
```

```
  color: #00b;
  text-decoration: none;
  font-size: 14px;
}

#demos a:hover {
  text-decoration: underline;
}

#demos tbody tr {
  border-top: 1px solid #DCDCDC;
}



#demos .demo p {
  margin-top: 0;
  margin-bottom: 5px;
}

#demos .support {
  width: 105px;
}

#demos .support span {
  cursor: pointer;
  overflow: hidden;
  float: left;
  display: block;
  height: 16px;
```

```
  width: 16px;
  text-indent: -9999px;
  background-image: url(/images/browsers.gif);
  background-repeat: none;
  margin-right: 5px;
}

#demos .support span.selected {
  outline: 1px dashed #75784C;
}

#demos .support span.safari {
  background-position: 0 0;
}

#demos .support span.chrome {
  background-position: 16px 0;
}

#demos .support span.firefox {
  background-position: 32px 0;
}

#demos .support span.ie {
  background-position: 48px 0;
}

#demos .support span.opera {
  background-position: 64px 0;
```

```
}

#demos .support span.nightly {
  opacity: 0.5;
  filter:alpha(opacity=50);
}

#demos .support span.none {
  opacity: 0.1;
  filter:alpha(opacity=10);
}

#demos .tags {
  width: 140px;
}

#demos .tags span {
  font-size: 11px;
  color: #6E724E;
  padding: 2px 5px;
  border: 1px solid #D7D999;
  background: #FFFFC6;
  -moz-border-radius: 10px;
  -webkit-border-radius: 10px;
  border-radius: 10px;
  cursor: pointer;
}

#demos .tags span:hover,
```

```
#demos .tags span.selected {
  border: 1px solid #75784C;
  background: #FF7;
  color: #333521;
}
```

14.2.2.2 *Multiviewstyle.css*

```
.header{
 width: 700px;
 height: 70px;
}

h1 {
font: "Helvetica Neue", Helvetica, sans-serif;
  padding-top: 10px;
}

 #web {
  float:right;

}
#logo{
position: absolute;
      top:100px;
      bottom: 60;
      left: 935px;
      width: 148px;
      height: 60px;
```

```
      padding: 0;

      margin: 0;

      border: none;

      background: transparent;

 }


#fondo{

    width:400px;

    top:100px;

    height:250px;

    background:url('Dateien/fondo.png') ;

    margin:140px auto 50px;

    position:absolute;

}
```

14.2.3 JAvascript Code for the Multi-view Scenario.

14.2.3.1 *H5utils script.*

```
// For discussion and comments, see:
http://remysharp.com/2009/01/07/html5-enabling-script/

/*@cc_on'abbr article aside audio canvas details figcaption figure
footer header hgroup mark menu meter nav output progress section
summary time
video'.replace(/\w+/g,function(n){document.createElement(n)})@*/

var addEvent = (function () {

  if (document.addEventListener) {

    return function (el, type, fn) {

      if (el && el.nodeName || el === window) {

        el.addEventListener(type, fn, false);

      } else if (el && el.length) {

        for (var i = 0; i < el.length; i++) {

          addEvent(el[i], type, fn);
```

```
        }
      }
    };
  } else {
    return function (el, type, fn) {
      if (el && el.nodeName || el === window) {
        el.attachEvent('on' + type, function () { return fn.call(el, 
window.event); });
      } else if (el && el.length) {
        for (var i = 0; i < el.length; i++) {
          addEvent(el[i], type, fn);
        }
      }
    };
  }
})();

(function () {
var pre = document.createElement('pre');
pre.id = "view-source"
// private scope to avoid conflicts with demos
addEvent(window, 'click', function (event) {
  if (event.target.hash == '#view-source') {
    // event.preventDefault();
    if (!document.getElementById('view-source')) {
      pre.innerHTML = ('<!DOCTYPE html>\n<html>\n' + 
document.documentElement.innerHTML + '\n</html>').replace(/[<>]/g, 
function (m) { return {'<':'&lt;','>':'&gt;'}[m]});
      document.body.appendChild(pre);
    }
```

```
    document.body.className = 'view-source';

    var sourceTimer = setInterval(function () {
      if (window.location.hash != '#view-source') {
        clearInterval(sourceTimer);
        document.body.className = '';
      }
    }, 200);
  }
});
  })();
```